

## เกมกลคนเดินช่องไม่ซ้ำ (no\_repeat)

เอราวัณกับบีโรเฟนเป็นเพื่อนซี้กัน วันหนึ่งบีโรเฟนท้าเอราวัณด้วยเกมสุดท้าทายที่มีชื่อว่า “เกมกลคนเดินช่องไม่ซ้ำ” ซึ่งมีกติกา ดังนี้

- กระดานเกมมีทั้งหมด  $N$  ช่อง โดยมีหมายเลขตั้งแต่ 0 ไปถึง  $N-1$  เอราวัณจะเริ่มต้นที่ช่องหมายเลข 0
- ในแต่ละตาเดินนั้น
  - สมมติว่าในขณะนี้ เอราวัณอยู่ในช่องที่  $P$
  - เอราวัณจะต้องเลือกตัวเลขจำนวนเต็ม  $D$  ในช่วง  $1..N-1$
  - จากนั้นบีโรเฟนจะเป็นคนเลือกจะให้เอราวัณไปลงช่องที่  $(P+D) \bmod N$  หรือ  $(P-D) \bmod N$
- เกมจะจบลงทันทีถ้าเอราวัณกลับมาแะช่องที่เคยเดินไปแล้ว ตอนเริ่มต้นถือว่าเอราวัณได้เดินแะช่องเริ่มต้น คือช่อง 0 แล้ว
- คะแนนที่เอราวัณได้รับคือจำนวนช่องที่เดินไม่ซ้ำ ซึ่งก็คือจำนวนก้าวที่เดินก่อนจะจบเกมนั่นเอง

## งานของคุณ

ให้คุณเขียนโปรแกรมเล่นเกมนี้โดยสวมบทบาทเป็นเอราวัณเพื่อแข่งกับเพื่อนผู้ฉลาดปราดเปรื่องและชั่วร้ายอย่างบีโรเฟน โดยคุณต้องพยายามทำแต้มให้มากที่สุดเท่าที่เป็นไปได้โดยที่บีโรเฟนจะเป็นคนขีดขวางพยายามให้คุณได้แต้มน้อยที่สุดแน่นอนว่าแต้มที่ได้จะขึ้นกับค่าของ  $N$  และตัวเลือกของบีโรเฟนด้วย

## Implementation

คุณมีหน้าที่เขียนสองฟังก์ชันเพื่อเล่นเป็นเอราวัณ คือ **initGame (N)** กับ **chooseD (P)** ซึ่งมีรายละเอียดดังนี้

- `void initGame(int N)`: คำสั่งนี้จะบอกเอราวัณให้ทราบถึงขนาดของกระดาน ( $N$ )
- `int chooseD(int P)`: ในแต่ละครั้งที่คำสั่งนี้ถูกเรียก เอราวัณจะทราบถึงช่องปัจจุบัน ( $P, 0 \leq P < N$ ) ของตานี้ ฟังก์ชันนี้มีหน้าที่บอกว่าในตานี้เอราวัณจะเลือกให้ค่า  $D$  เป็นเท่าไร จงอย่าลืมว่า  $1 \leq D < N$

คำสั่ง `initGame (N)` จะถูกเรียกเพียงครั้งเดียวก่อนเริ่มเกม หลังจากนั้นฟังก์ชัน `chooseD (P)` จะถูกเรียกตาละครั้งเรื่อยๆ จนกว่าเกมจะจบลง โดยครั้งแรกที่ `chooseD (P)` ถูกเรียกนั้นรับประกันว่า  $P=0$

## ตัวอย่างการทำงาน

คำสั่งพร้อม arguments	ค่าที่คืน	คำอธิบาย
initGame (5)	(void)	เริ่มต้นให้กระดานมีจำนวน 5 ช่องคือ {0, 1, 2, 3, 4}
chooseD (0)	2	เอราวัณเลือก $D = 2$ ในตอนนี้ บีโรเฟนเลือกได้ว่าจะให้เอราวัณไปลงช่องที่ 2 หรือ 3 ในกรณีนี้ บีโรเฟนตัดสินใจเลือกช่องหมายเลข 3 เป็นปลายทาง (หมายความว่า {0, 3} ถูกแหวะแล้ว)
chooseD (3)	1	เอราวัณเลือก $D = 1$ บีโรเฟนเลือกไปลงช่องหมายเลข 2 ({0, 2, 3} ถูกแหวะแล้ว)
chooseD (2)	3	เอราวัณเลือก $D = 3$ บีโรเฟนเลือกไปลงช่องหมายเลข 0 ซึ่งถูกแหวะไปแล้ว เกมจึงจบลงด้วยคะแนน 3 คะแนน ( $ \{0, 2, 3\}  = 3$ )

## การให้คะแนน

คะแนนของโปรแกรมของคุณจะได้มาจากการเทียบแต้มสำหรับชุดทดสอบนั้นกับแต้มของผู้เข้าแข่งขันคนอื่นและแต้มของกรรมการ สมมติว่าชุดทดสอบนั้นมีคะแนนเต็ม  $S$  คะแนน แต้มของคุณคือ  $P$  และแต้มที่สูงที่สุดของผู้เข้าแข่งขันทุกคนรวมทั้งของกรรมการคือ  $P_{max}$  จะได้ว่าคะแนนของคุณคือ  $S * (P / P_{max})^2$  โดยจะปัดลง (round down) เป็นจำนวนเต็ม

สำหรับตัวอย่างของคะแนนที่ได้นั้น ถ้า  $S = 5$  และ  $P_{max} = 18$  และ  $P$  มีค่าต่างๆ คะแนนของคุณจะเป็นดังนี้

$P$	2	5	8	11	13	15	16	17	18
คะแนน	0	0	0	1	2	3	3	4	5

โดยระหว่างการแข่งขันนั้น คุณจะไม่สามารถทราบคะแนนที่ได้ แต่จะทราบเพียงแค่แต้มที่ได้จากเกมของชุดทดสอบนั้นๆ แต่หลังจากหมดเวลาแล้ว แต้มของทุกคนจะถูกนำมาเทียบกันเพื่อคำนวณคะแนน

ลักษณะและขอบเขตของชุดข้อมูลทดสอบมีดังนี้

- สำหรับทุกชุดข้อมูลทดสอบ:  $2 \leq N \leq 2^{20}$  โดยที่  $N = M * 2^K$ ,  $1 \leq M \leq 1000$ ,  $0 \leq K \leq 20$  และ  $N, M, K$  เป็นจำนวนเต็มทั้งสามตัว
- ชุดทดสอบทั้งหมดจะแบ่งเป็น 6 กลุ่มดังต่อไปนี้ โดยแต่ละชุดทดสอบย่อย คะแนนจะคิดแยกจากกัน
- กลุ่มที่ 1 (ไม่เกิน 15 คะแนน):  $M = 1, K \leq 10$
- กลุ่มที่ 2 (ไม่เกิน 15 คะแนน):  $M = 1, K \leq 20$
- กลุ่มที่ 3 (ไม่เกิน 20 คะแนน):  $M \leq 5$
- กลุ่มที่ 4 (ไม่เกิน 20 คะแนน):  $M \leq 15$
- กลุ่มที่ 5 (ไม่เกิน 20 คะแนน):  $M \leq 100$
- กลุ่มที่ 6 (ไม่เกิน 10 คะแนน):  $M \leq 1000$

ถ้าโปรแกรมของคุณใช้เวลาในการทำงานมากเกินไปจนทำงานช้าหรือคืนค่า  $D$  ออกนอกช่วงที่กำหนด ฯลฯ แต้มที่ได้จากชุดทดสอบนั้นจะเป็น 0 ทันที

#### ข้อจำกัดการทำงาน

Time limit: 3 seconds

Memory limit: 256 MB

หมายเหตุ: เนื่องจากโปรแกรมของคุณจะใช้ทรัพยากรร่วมกับโปรแกรมเล่นเกมของบีโชน คุณควรระวังอย่าใช้ทรัพยากรใกล้ขีดจำกัดจนเกินไป แม้ว่าโปรแกรมของบีโชนจะประหยัดเวลาและความจำแค่ไหนก็ตาม