

ช็อกโกแลต

มีช็อกโกแลตแท่งหนึ่ง เป็นช็อกโกแลตแบบแท่งที่แบ่งเป็นชิ้นเล็ก ๆ จำนวน n ชิ้นย่อย (เช่นรูปด้านขวานี้เป็นช็อกโกแลตที่แบ่งเป็น 6 ชิ้นย่อย)



เราต้องการแบ่งกินช็อกโกแลตนี้ไปหลาย ๆ วัน โดยแต่ละวันเราจะเลือกหักช็อกโกแลตออกมาจากปลายด้านหนึ่งเป็นจำนวนไม่กี่ชิ้น แล้วกินให้หมด อย่างไรก็ตาม เราตั้งกฎกับตัวเองขึ้นมาว่า จำนวนชิ้นที่จะกินต่อวันนั้นจะต้องเป็นจำนวนเต็มที่อยู่ใน set S เท่านั้น

ในแต่ละวันที่เรากินช็อกโกแลตนั้น เราจะจดบันทึกจำนวนชิ้นที่กินลงบนกระดานด้วย ตัวอย่างเช่น หากให้ S เป็น $\{1, 2, 3\}$ วิธีการกินช็อกโกแลตแบบหนึ่งที่เป็นไปได้จะทำให้เราจดตัวเลขเป็น $[1, 2, 1, 2]$ หรืออีกแบบที่เป็นไปได้คือ $[1, 2, 2, 1]$ หรืออีกแบบ เช่น $[3, 1, 2]$ ด้วยความเป็นนิสิตภาควิชาวิศวกรรมคอมพิวเตอร์ เราก็เลยอยากทราบว่าเรามีวิธีการแบ่งกินช็อกโกแลตนี้จนหมดแท่งได้ทั้งหมดกี่แบบที่แตกต่างกัน

ข้อมูลนำเข้า

- บรรทัดแรกประกอบด้วยจำนวนเต็ม 2 ตัวคือ n และ k ซึ่งระบุจำนวนชิ้นของช็อกโกแลต $2 \leq n \leq 10^4$ และ ระบุขนาดของเซต S ($1 \leq k \leq 10$)
- บรรทัดที่สองประกอบด้วยจำนวนเต็ม k ตัว ซึ่งระบุค่าของเซต S รับประกันว่า ค่าใน S นั้นไม่ซ้ำกันเลย และรับประกันว่ามี 1 เป็นสมาชิกของ S แน่แน่นอน และค่าใน S นั้นจะไม่มากกว่า n

ข้อมูลส่งออก

มีหนึ่งบรรทัดประกอบด้วยจำนวนเต็มหนึ่งตัว ซึ่งระบุจำนวนรูปแบบที่เป็นไปได้ทั้งหมด อย่างไรก็ตาม ค่าดังกล่าวอาจจะมีค่าสูงมาก ๆ จนไม่สามารถเก็บในตัวแปรแบบ `int` หรือ `long` ได้ ดังนั้น ผลลัพธ์ที่ตอบ ให้ตอบผลลัพธ์ในรูปแบบ `mod 1000003` (คือหารผลลัพธ์ด้วย 1,000,003 แล้วตอบเฉพาะเศษของการหาร)

(สมมติว่า เรามีตัวแปร x ที่เอาไว้เก็บคำตอบ แล้วเราต้องมีการเพิ่มค่า x ด้วย y เราสามารถทำได้โดยใช้คำสั่ง $x = (x + y \% 1000003) \% 1000003$ เพื่อคำนวณค่า x ให้อยู่ในรูปแบบ `mod 1000003` ได้)

ตัวอย่าง

ข้อมูลนำเข้า	ข้อมูลส่งออก
4 2 1 3	3 // ทั้งสามแบบ คือ $[1, 3]$, $[3, 1]$ และ $[1, 1, 1, 1]$
10 2 1 2	89
10 5 1 2 4 6 8	196
999 5 1 2 4 50 999	450619

ข้อมูลทดสอบ

- 20% ของชุดข้อมูลทดสอบ จะมีค่า S เป็น 1 และ 2 เสมอ
- 80% ของชุดข้อมูลทดสอบ ไม่มีข้อกำหนดอื่นใด