

OR Operator

[Time limit : 1s] [Memory limit : 64 MB]

เมื่อคุณได้เล่นเกมเกี่ยวกับลำดับของจำนวนเต็ม ในตอนเริ่มต้นคุณจะได้ลำดับหนึ่งลำดับที่มีสมาชิก N ตัว และคุณสามารถแบ่งลำดับที่คุณได้รับให้เป็นลำดับย่อยๆได้ไม่เกิน K ลำดับโดยมีเงื่อนไขดังนี้

- สำหรับแต่ละลำดับย่อยจะต้องมีสมาชิกอย่างน้อย 1 ตัว
- การแบ่งลำดับย่อยเสมือนเป็นการตัดลำดับโดยอันดับของสมาชิกในลำดับจะคงเดิมไม่สามารถเปลี่ยนได้ เช่น $(3\ 4\ 2\ 3\ 1)$ สามารถตัดให้เป็นลำดับย่อย 2 ลำดับได้ 4 แบบคือ $(3)\ (4\ 2\ 3\ 1)$, $(3\ 4)\ (2\ 3\ 1)$, $(3\ 4\ 2)\ (3\ 1)$ และ $(3\ 4\ 2\ 3)\ (1)$

ซึ่งกติกาของเกมมีอยู่ว่า ลำดับใดๆจะมีค่าคงที่ S ซึ่งมีค่าเท่ากับผลรวมของสมาชิกทุกตัวในลำดับ และ นิยามค่า Q มีค่าเท่ากับ $S_1\ |\ S_2\ |\ S_3\ |\ \dots\ |\ S_k$ โดย S_i คือค่าคงที่ S ของลำดับย่อยที่ i และ $|$ คือการทำ Bitwise Operator OR คุณต้องการหาว่าคุณสามารถหาค่า Q ได้น้อยที่สุดเท่าใดเมื่อคุณสามารถตัดลำดับที่ได้รับเป็น ลำดับย่อยได้ไม่เกิน K ลำดับ

Bitwise Operator OR คือ การนำเลขฐาน 10 แปลงเป็นเลขฐาน 2 และนำตัวเลขในแต่ละหลักมาทำการ OR กันโดย เลข 1 , 0 ในระบบฐานสองจะเทียบได้กับ นิพจน์ที่มีค่าความจริงเป็น **จริง** และ **เท็จ** ตามลำดับ แล้วจึงทำการแปลงกลับเป็นเลขฐาน 10 เช่น $5\ |\ 13$ แปลงเป็นระบบเลขฐาน 2 คือ $00101_2\ |\ 01101_2 = 01101_2$ ซึ่งมีค่าเท่ากับ 13 ดังนั้น $5\ |\ 13 = 13$

ข้อมูลนำเข้า

บรรทัดแรกประกอบด้วย จำนวนเต็ม N , K ($1 \leq K \leq N \leq 2,000$) แสดงถึงจำนวนสมาชิกในลำดับ และ จำนวนลำดับย่อยมากที่สุดที่สามารถแบ่งได้

บรรทัดที่สองประกอบด้วย จำนวนเต็ม N จำนวน มีค่าตั้งแต่ 1 ถึง 1,000,000,000 แสดงถึงสมาชิกของลำดับในตำแหน่งต่างๆ

ข้อมูลส่งออก

ประกอบด้วยจำนวนเต็มหนึ่งตัว แสดงถึง Q น้อยที่สุดที่เป็นไปได้เมื่อคุณสามารถตัดลำดับที่ได้รับเป็น ลำดับย่อยได้ไม่เกิน K ลำดับ

ตัวอย่าง

Input	Output
6 3 8 1 2 1 5 4	11
8 4 1 2 1 1 1 0 4 6	6

คำอธิบายตัวอย่าง

ตัวอย่าง 1 เราสามารถแบ่ง (8 1 2 1 5 4) เป็น (8 1 2) (1) (5 4) และ $Q = 11 \mid 1 \mid 9 = 11$

ตัวอย่าง 2 เราสามารถแบ่ง (1 2 1 1 1 0 4 6) เป็น (1 2 1 1 1) (0) (4) (6) และ $Q = 6 \mid 0 \mid 4 \mid 6 = 6$

