

กำแพงหนึ่งแบ่งเป็นช่อง ๆ ไว้  $N$  ช่อง ( $1 \leq N \leq 100,000$ ) แต่ละช่องระบายสีไว้หลากหลาย สำหรับช่องที่  $i$  เมื่อ  $1 \leq i \leq N$  สีที่ระบายคือสีหมายเลข  $C_i$  ( $1 \leq C_i \leq 100,000$ )

ส่วนของกำแพงที่ต่อเนื่องกันจะดูละลานตา ถ้ามีสีที่แตกต่างกัน อย่างน้อย  $K$  สี ( $1 \leq K \leq N$ ) คุณต้องการหาว่ามีส่วนของกำแพงที่ต่อเนื่องกันกี่ส่วนที่ดูละลานตา

พิจารณาตัวอย่างที่  $N = 8$  และ  $K = 3$  ดังต่อไปนี้ ด้านล่างตารางแสดงตัวอย่างกำแพงและสีที่ทาในแต่ละช่อง

1	2	1	3	4	3	1	3
---	---	---	---	---	---	---	---

ส่วนของกำแพงที่ต่อเนื่องกันและดูละลานตามีทั้งหมด 18 ส่วน ดังต่อไปนี้

1	2	1	3	1	3	4						
1	2	1	3	4	1	3	4	3				
1	2	1	3	4	3	1	3	4	3	1		
1	2	1	3	4	3	1	1	3	4	3	1	3
1	2	1	3	4	3	1	3	3	4	3	1	
2	1	3	3	4	3	1	3					
2	1	3	4	4	3	1						
2	1	3	4	3	4	3	1	3				
2	1	3	4	3	1							
2	1	3	4	3	1	3						

หมายเหตุ: มีข้อมูลทดสอบ 50% ที่  $N \leq 100$

### ข้อมูลนำเข้า

บรรทัดแรกระบุจำนวนเต็มสองจำนวน  $N$  และ  $K$  ( $1 \leq N \leq 100,000$ ;  $1 \leq K \leq N$ ) มีข้อมูลทดสอบ 50% ที่  $N \leq 100$

อีก  $N$  บรรทัดจะระบุสีของช่องกำแพง กล่าวคือในบรรทัดที่  $1+i$  สำหรับ  $1 \leq i \leq N$  จะระบุจำนวนเต็ม  $C_i$  ที่เป็นจำนวนเต็มบวกแทนหมายเลขสีในช่องกำแพงช่องที่  $i$  ( $1 \leq C_i \leq 100,000$ )

### ข้อมูลส่งออก

มีหนึ่งบรรทัด เป็นจำนวนเต็มหนึ่งจำนวนระบุจำนวนส่วนของกำแพงที่ต่อเนื่องกันที่ดูละลานตา

หมายเหตุ: ในกรณีทดสอบที่  $N$  มีค่ามาก คำตอบอาจจะเป็นจำนวนเต็มขนาดใหญ่ ผู้ใช้ภาษา C/C++ อาจจะต้องใช้ตัวแปรประเภท long long ในการเก็บค่า

**เงื่อนไขการทำงาน** โปรแกรมภาษา C/C++ ต้องทำงานภายใน 1.5 วินาที ภาษา Python ทำงานในเวลา 3 วินาที ใช้หน่วยความจำไม่เกิน 256 MB

(ตัวอย่างข้อมูลทดสอบและรายละเอียดการให้คะแนนอยู่หน้าถัดไป)

### การให้คะแนน

- มีข้อมูลทดสอบ 50% ที่  $N \leq 100$
- ข้อมูลทดสอบอีก 50% ที่เหลือ  $N$  จะมีขนาดใหญ่ โปรแกรมที่เขียนจะต้องมีประสิทธิภาพมากพอที่จะทำงานได้ในเวลาที่กำหนด ในส่วนนี้ โปรแกรมที่ส่งจะต้องทำงานถูกต้องในทุกกรณีทดสอบกลุ่มนี้จึงจะได้คะแนน 50%

### ตัวอย่าง

Input	Output
8 3 1 2 1 3 4 3 1 3	18