

List Rearrangement (version 2)

มีลำดับของของจำนวนเต็มหลายจำนวน (D) เช่น $[8, 3, 1, 6]$ เราต้องการสร้างลำดับใหม่ (X) ที่เป็นการหยิบข้อมูลจากลำดับ D มาทีละตัว การหยิบตัวถัดไปจะถูกกำหนดโดยลำดับของข้อมูลการหยิบ (P) โดยในการหยิบข้อมูลตัวถัดไปจะไม่สนใจข้อมูลที่เคยหยิบไปแล้ว (เสมือนว่าไม่มีข้อมูลนั้นอยู่แล้ว)

จากข้อกำหนดที่กล่าวมา สามารถนำมาเขียนเป็นข้อกำหนดที่เป็นภาษาทางการได้ดังต่อไปนี้

- ให้ D เป็นลำดับของจำนวนเต็มไม่ติดลบ d_1, d_2, \dots, d_n ในลำดับนี้ อาจมีข้อมูลบางตัวเป็นตัวยกห้าม
ขอแสดงตัวยกห้ามด้วยตัวสีแดง เช่น $D = 7, 8, 9, 6, 4, 2, 3, 4$
- ให้ $D^*(k)$ คือ ข้อมูลใน D เริ่มที่ซ้ายสุดไล่ไปทางขวาเรื่อย ๆ (ข้ามตำแหน่งตัวยกห้าม) จนเจอตัวที่ k ถ้าเลยตัวขวาสุดของ D ให้วกกลับมาทางซ้ายสุด เช่น $D = 7, 8, 9, 6, 4, 2, 3, 4$ จะได้ $D^*(1)$ คือ 7, $D^*(3)$ คือ 4, $D^*(9)$ คือ 3
- ให้ P เป็นลำดับของจำนวนที่สร้างจาก D โดยจะเป็นเริ่มต้นที่ d_n แล้วหยิบจากขวามาซ้าย เพิ่มต่อท้ายลิสต์ เพิ่มด้านหน้าลิสต์ ... สลับแบบนี้ไปเรื่อย ๆ
เช่น D คือ $7, 8, 9, 5, 4, 2, 3, 4$ จะได้ P คือ $8, 5, 2, 4, 3, 4, 9, 7$

สิ่งที่ต้องการให้หาคือ ลำดับใหม่ X ซึ่งเป็นลำดับของจำนวนเต็ม x_1, x_2, \dots, x_n โดยที่

- เริ่มจาก D ที่ไม่มีตัวยกห้าม
- ให้ x_1 คือ $D^*(1+p_1)$ และให้ตัวที่หาได้นี้เป็นตัวยกห้ามใหม่ใน D
- ให้ x_2 คือ $D^*(1+p_1+p_2)$ และให้ตัวที่หาได้นี้เป็นตัวยกห้ามใหม่ใน D
- ให้ x_3 คือ $D^*(1+p_1+p_2+p_3)$ และให้ตัวที่หาได้นี้เป็นตัวยกห้ามใหม่ใน D
- ...
- ให้ x_n คือ $D^*(1+p_1+p_2+\dots+p_n)$ ของ D

ตัวอย่าง D คือ $8, 3, 1, 6$ เรียงลำดับได้ p_1, p_2, p_3, p_4 คือ $3, 6, 1, 8$ และหาลำดับ x_1, x_2, x_3, x_4 ได้ดังนี้

ขั้นตอน	ผลลัพธ์ X
$D = 8, 3, 1, 6$ $x_1 = D^*(1+p_1) = D^*(1+3) = D^*(4)$ $8 \rightarrow 3 \rightarrow 1 \rightarrow 6$	6
$D = 8, 3, 1, 6$ $x_2 = D^*(1+p_1+p_2) = D^*(1+3+6) = D^*(10)$ $8 \rightarrow 3 \rightarrow 1 \rightarrow 8 \rightarrow 3 \rightarrow 1 \rightarrow 8 \rightarrow 3 \rightarrow 1 \rightarrow 6$	6, 8
$D = 8, 3, 1, 6$ $x_3 = D^*(1+p_1+p_2+p_3) = D^*(1+3+6+1) = D^*(11)$ $3 \rightarrow 1 \rightarrow 3 \rightarrow 1 \rightarrow 3 \rightarrow 1 \rightarrow 3 \rightarrow 1 \rightarrow 3 \rightarrow 1 \rightarrow 3 \rightarrow 6$	6, 8, 3
$D = 8, 3, 1, 6$ $x_4 = D^*(1+p_1+p_2+p_3+p_4) = D^*(1+3+6+1+8) = D^*(19)$ $1 \rightarrow 1 \rightarrow 1 \rightarrow \dots \rightarrow 1$	6, 8, 3, 1

ให้เขียนฟังก์ชัน `find_P(d)` และ `rearrange(d)` เพื่อให้โปรแกรมทำงานได้ตามข้อกำหนดข้างบน

```
def find_P(D):  
    # D เป็นลำดับของจำนวนเต็มไม่ติดลบ  
    # คืนค่าเป็นลำดับของจำนวน P ตามข้อกำหนดข้างบน  
    ???  
  
def rearrange(D):  
    # D เป็นลำดับของจำนวนเต็มไม่ติดลบ  
    # คืนค่าลำดับใหม่ X ตามข้อกำหนดข้างบน  
    ???  
  
exec(input().strip()) # DON'T remove this line
```

ข้อมูลนำเข้า

คำสั่งภาษา Python ที่ต้องการให้ทำงาน

ข้อมูลส่งออก

ผลที่ได้จากการสั่งทำงานคำสั่งที่ได้รับ

ตัวอย่าง

input (จากแป้นพิมพ์)	output (ทางจอภาพ)
<code>print(*find_P([8,3,1,6]))</code>	3 6 1 8
<code>print(*find_P([8,3,1,6,3,4,1]))</code>	8 1 3 1 4 6 3
<code>print(*rearrange([999]))</code>	999
<code>print(*rearrange([9,9,9]))</code>	9 9 9
<code>print(*rearrange([8,3,1,6]))</code>	6 8 3 1
<code>print(*rearrange([6,5,4,3,2,1]))</code>	1 3 5 2 4 6
<code>print(*rearrange([2,2,2,2,3,0,0,3,2,2]))</code>	2 0 0 2 3 2 2 2 2 3
<code>print(*rearrange([4,339389,2099,3039392,1292832,4,8]))</code>	1292832 3039392 4 8 339389 4 2099

หมายเหตุ: ให้ `D` เป็นลิสต์ `[1, 2, 3, 4]` คำสั่ง `print(*D)` จะแสดงข้อมูลทุกตัวในลิสต์ออกมาโดยไม่มีเครื่องหมายใด ๆ เป็น `1 2 3 4`