

# Dokumentáció

## Feladatléírás

Kapsz egy tömböt, amely minden  $1 \leq i \leq n$  közötti számot pontosan egyszer tartalmaz. Az Ön feladata, hogy növekvő sorrendben gyűjtse össze a számokat 1-től n-ig.

Minden körben balról jobbra haladva végigmész a tömbön, és összegyűjtöd a lehető legtöbb számot. Mennyi lesz a körök száma összesen?

## Bemenet

Az első sorban van egy  $n$  egész szám: a tömb mérete.

A következő sor  $n$  egész számot tartalmaz  $x_1, x_2, \dots, x_n$ : a tömbben lévő számok.

## Kimenet

Nyomtatson ki egy egész számot: a fordulók számát.

## Korlátozások

$$1 \leq n \leq 2 \cdot 10^5$$

## Példa

Bemenet:

5

4 2 1 5 3

Kimenet:

3

---

## Formula

A megoldás a következő lépésekre épül:

- Indexek összegyűjtése: Először megkeressük az 1-től  $n!$  terjedő számok helyét a tömbben, hogy azokat növekvő sorrendben tudjuk elemezni.
- Diszjunkt szakaszok azonosítása: A számokat azok eredeti sorrendjében vizsgáljuk. Ha egy szám indexe kisebb, mint az előző számé, új szakasz kezdődik, amely egy új kört jelent.

---

## Lépések

- Bemenet feldolgozása: Az algoritmus beolvassa a tömb elemeit és az 1-től  $n!$  terjedő számok indexeit.
- Sorrend elemzése: Az indexeket növekvő sorrendben vizsgáljuk, hogy azonosítsuk, hány különálló szakasz van a tömbben.

3. Körök számának meghatározása: Minden új szakasz egy újabb körnek felel meg. A körök számát növeljük, valahányszor az aktuális szám indexe kisebb, mint az előző szám indexe.

### **Algoritmus Hatékonysága**

Az algoritmus lineáris időben fut, mivel csak egyszer kell végig haladnunk a tömbön az indexek meghatározásához és egyszer az indexek sorrendjének elemzéséhez.

---

### **Implementáció**

A program a Python nyelven került megvalósításra, és a bemenetek standard inputról érkeznek. A kimenet egyetlen számot tartalmaz, amely a szükséges körök számát adja meg.