

Dobozok

Van N különböző méretű dobozunk. Bármely kisebb méretű berakható bármely nagyobb méretűbe. Kezdetben bizonyos dobozok benne vannak más dobozokban, de mindegyikben legfeljebb egy másik doboz lehet közvetlenül. Azt mondjuk, hogy egy doboz szabad, ha nem tartalmazza másik doboz. A dobozok egy helyzete leírható egy D_1, \dots, D_N számsorozattal ahol $(0 \leq D_i \leq N)$. $D_i = 0$ azt jelenti, hogy az i . doboz szabad, egyébként D_i annak a doboznak a sorszáma, amelyik közvetlenül tartalmazza az i . dobozt. Adott kiindulási helyzetből ki kell alakítani egy másik helyzetet. Egy lépésben kétféle műveletet végezhetünk:

1. Egy szabad dobozt (a benne lévőekkel együtt) berakjuk egy üres szabad dobozba.
2. Egy nem üres szabad dobozt felnyitunk és kivesszük belőle a benne lévő dobozt (az abban lévőekkel együtt).

Írj programot, amely kiszámítja, hogy legkevesebb hány lépéssel alakítható ki a kezdeti helyzetből a kívánt cél helyzet!

Bemenet

A *standard bemenet* első sorában a dobozok száma van ($2 \leq N \leq 100\,000$). A második sor N egész számot tartalmaz, a kezdeti helyzet leírását. A harmadik sor is N egész számot tartalmaz, a cél helyzet leírását. A bemenetre teljesül, hogy mindkét helyzet valódi.

Kimenet

A *standard kimenet* első és egyetlen sorába azon legkevesebb lépések számát kell írni, amellyel a kezdeti helyzetből kialakítható a cél helyzet!

Példa

bemenet	kimenet
6	3
2 5 4 0 0 0	
2 6 4 5 0 0	

Korlátok

Időlimit: 0.1 mp.

Memórialimit: 32 MB

A tesztek 30%-ában $N \leq 100$.