

Round Table

We have a round table with $2 \cdot N$ seats ($1 \leq N \leq 2000$). The seats are numbered with the numbers from 1 to $2 \cdot N$ in order. This round table is in a round room with two doors. Door 1 is between seats 1 and $2N$ on the table, and door 2 is between seats N and $N+1$. We have $2N$ guests that we want to seat. Each guest has a id from 1 to $2N$ and we want to seat him/her exactly at the seat corresponding to his id. All the guest are split into two groups of N and each group is waiting in front of a door in a line. This room unfortunately has the problem that if you let a person go from door 1 and he/she has to seep at position p , to get to his seat he has to make either all people seating on seats from 1 to $p-1$ or all people seating on seats $p+1$ to $2N$ to get up to let him in. This presents you with the problem of how to let the guests in to cause minimum number of getting ups. The only thing you control is from which door to let the next guest. The guests that are waiting in front of door 1 can only get in from door 1 the same is true with door 2.

Input

On the first line there will one integer - N . Two more lines of input follow each with N integers in the range $[1, 2N]$ - the guests waiting in front of door 1 and door 2 respectively. The first people in the list is the first to enter the room.

Output

Single integer the total min number of stand-ups that will happen if we choose the best sequence of letting guest in.

Example

Input:

```
3
4 5 3
6 2 1
```

Output:

```
3
```