Problem M

Moon Dancers

The moon dancers, as their name suggests, is a group of people who dances to pray the moon. The moon is a goddess for their tribe, they dance every full moon to get the moon blessings and guarantee the tribe another moon cycle of wellness.

Some researchers are very interested in the peculiar dance these moon dancers perform on the full moon. It is marvelous to see how they dance around a circle, each dancer positioned and performing an exact set of moves that have been passed between dancers for generations. At some point of the dance, each of the N dancers sit at some point in the perimeter of the circle, no two dancers sit in the same place, and they start singing the song of times. Then, suddenly, a set of K dancers stand up and dance rotating counter clockwise around the circle until each of the K dancers meet with a dancer that is sitting, making K pairs of dancers, all of the K dancers will have rotated the same amount of degrees around the circle. At this point the K pairs continue with the dance while the N-2*K (possibly none) dancers that are not matched continue singing the song of times.

Knowing your programming skills, researchers have come to you and ask for help on a very specific task: given the position of each of the N dancers when they sit at the perimeter of the circle, can you find the maximum number of pairs that can be matched to continue the dance?

Input

The first line contains a single integer N ($2 \le N \le 360$), representing the number of dancers in the tribe. The second and last line contains N integer numbers separated by a space, representing the angle in degrees a_i ($0 \le a_i < 360$) where the i-th dancer sits, it is guaranteed no two dancers will sit in the same place.

Output

Output a line containing a single integer. The maximum number of pairs that can be matched to continue the dance.

Input example 1	Output example 1
2	1
0 1	
Input example 2	Output example 2
6	3
0 1 2 90 91 92	
Input example 3	Output example 3
3	1
359 0 1	