

카드 섞기

백준 1091 문제

문제

문제

지민이는 카지노의 딜러이고, 지금 3명의 플레이어(0, 1, 2)가 있다. 이 게임은 N 개의 카드를 이용한다. ($0 \sim N-1$ 번)

일단 지민이는 카드를 몇 번 섞은 다음에, 그것을 플레이어들에게 나누어 준다. 0번째 위치에 있던 카드가 플레이어 0에게 가고, 1번째 위치에 있던 카드는 플레이어 1에게 가고, 2번째 위치에 있던 카드는 플레이어 2에게 가고, 3번째 위치에 있던 카드는 플레이어 0에게 가고, 이런식으로 카드를 나누어 준다. 하지만, 지민이는 약간 사기를 치려고 한다.

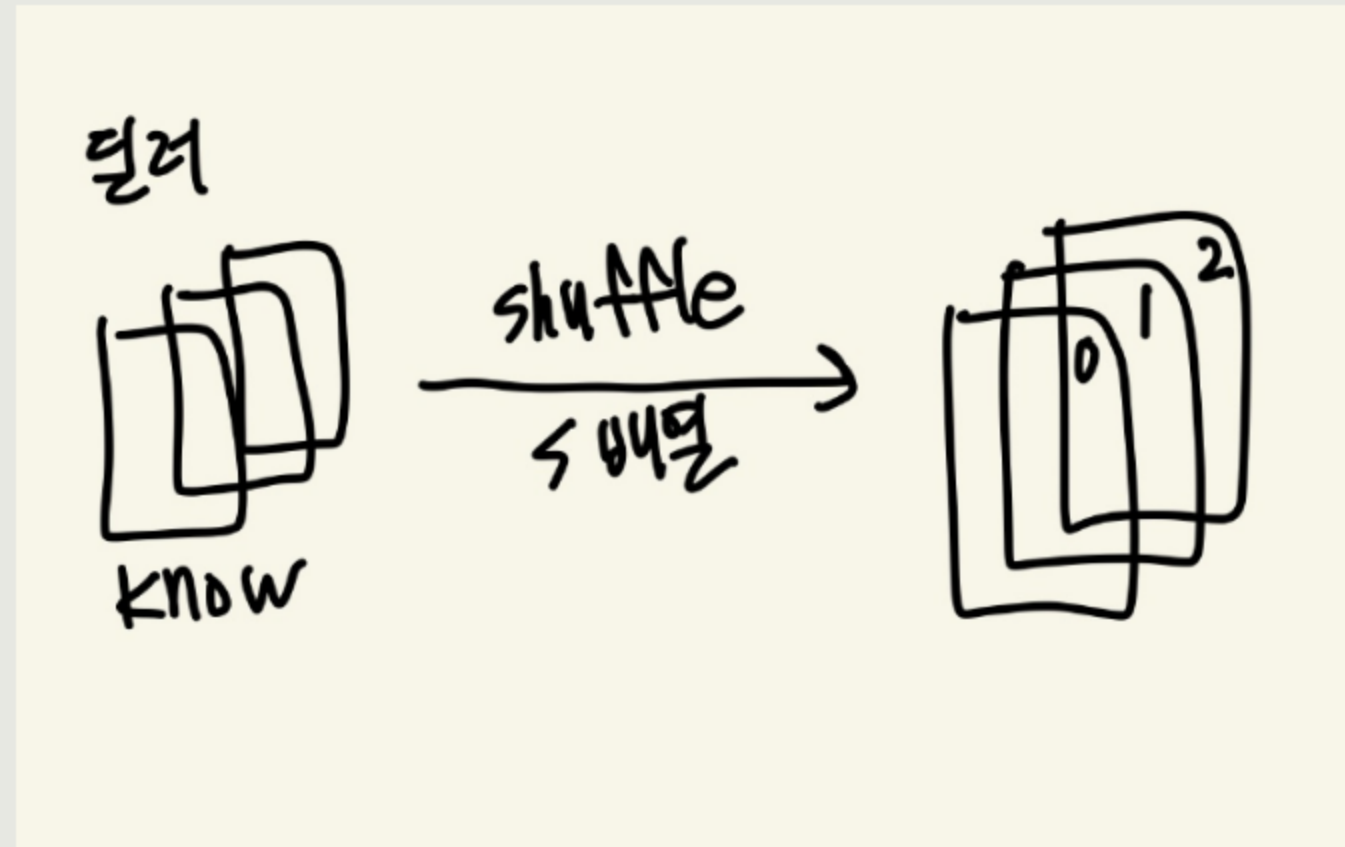
지민이는 처음에 카드를 섞기 전에 카드의 순서를 알고 있고, 이 정보를 이용해 각 카드가 특정한 플레이어에게 보내지게 할 것이다.

카드를 한 번 섞을 때는 주어진 방법을 이용해서만 섞을 수 있고, 이 방법은 길이가 N 인 수열 S 로 주어진다. 카드를 한 번 섞고 나면 i 번째 위치에 있던 카드는 $S[i]$ 번째 위치로 이동하게 된다.

각 카드가 어떤 플레이어에게 가야 하는지에 대한 정보는 길이가 N 인 수열 P 로 주어진다. 맨 처음 i 번째 위치에 있던 카드를 최종적으로 플레이어 $P[i]$ 에게 보내야한다.

지민이가 목적을 달성하기 위해 필요한 카드 섞는 횟수의 최솟값을 구하는 프로그램을 작성하시오.

문제



섞기 전 상태인 p배열에서 p[i] 유저에게 i번째 카드를 보내고 싶어함
이 때, 섞는 최소 횟수를 알고싶음

→ p배열의 카드 상태에서 시작하여 s배열의 정보로 카드를 섞은 뒤, 카드를 0, 1, 2, ...의
순서대로 나누어 지는 결과가 나온다면 카드를 섞어서 결과가 나올 수 있는 카드 순서임

입력 및 출력

입력

첫째 줄에 N 이 주어진다. N 은 3보다 크거나 같고, 48보다 작거나 같은 3의 배수이다.

둘째 줄에 길이가 N 인 수열 P 가 주어진다. 수열 P 에 있는 수는 0, 1, 2 중 하나이다.

셋째 줄에 길이가 N 인 수열 S 가 주어진다. 수열 S 에 있는 수는 모두 $N-1$ 보다 작거나 같은 자연수 또는 0이고 중복되지 않는다.

출력

첫째 줄에 몇 번 섞어야 하는지 출력한다. 만약, 섞어도 섞어도 카드를 해당하는 플레이어에게 줄 수 없다면, -1을 출력한다.

구현

p 배열의 카드를 s에 적힌대로 섞으면서
이미 섞었던 카드 정보가 나올 수 있음

-> 집합을 사용하여 카드를 추가해주면서 집합의 길이가 늘지 않았을 경우(중복된 카드 배열이 존재하는 경우) 반복문을 탈출하여 -1을 출력

알고리즘

```
N = int(input())

P = list(map(int, input().split()))
S = list(map(int, input().split()))

def listToTuple(list):
    return tuple(n for n in list)

# 카드를 섞은 뒤, 차례로 나누어 줌
goal = listToTuple([0, 1, 2]*(N//3))
```

카드 수 n장

섞기 전 원하는(보내고 싶은) 카드 상태
카드 섞을 정보

집합에 튜플의 자료구조로 넣을 예정

최종적으로 나누어 주는 카드

알고리즘

```
visited = set()
visited.add(listToTuple(P))

size = 0
while size < len(visited) and goal not in visited:
    move = [0]*N

    for i in range(N):
        move[S[i]] = P[i]

    P = move
    visited.add(listToTuple(P))

    size += 1

if goal in visited:
    print(size)
else:
    print("-1")
```

visited 집합 생성

집합에 p카드 상태 추가

길이 정보 size

길이가 줄어들지 않고 목표한 카드 배열이 없을 동안

카드를 섞는 과정을 반복함

반복문을 탈출하였을 경우

목표 카드 배열이 있다면 size 출력

그렇지 않다면 -1

감사합니다