C6. Cards Game

21500800 홍순규, 21600212 남진우

문제 해석

카드를 오름차순으로 받다가, 언제 내림차순으로 받아야 최댓값이 되는가.

아이디어

전체 카드 배열의 원소 각 각에 대해 왼쪽에서 오름차순으로 얻을 수 있는 최댓값과, 오른쪽에서 내림차순으로 얻을 수 있는 최댓값을 합쳐서 비교한다.

최댓값을 구하는 과정에서 겪은 시행 착오

brute force로 최댓값을 구해도 충분할 것 같아 시도했으나, Time Limit으로 실패했다.

스택과 DP를 이용한 방법으로 최적화하여 개선하려 했지만, 반례를 찾아 실패했다.

그 후 문제의 해결 방법을 고민하던 중, 부분 최대 증가 수열이라는 어디서 들어본 것 같은 이름을 떠올렸다.

그것을 인터넷에서 찾아보니 LIS라는 유명한 문제였고, brute force로 풀었다고 생각한 것이 전형적인 O(N^2) DP 풀이라는 것을 알 수 있었다.

또한, 이진 탐색으로 Lower bound를 찾아 LIS를 해결하는 O(NlogN)의 더 나은 풀이를 이해할 수 있었다.

풀이

LIS를 구하는 과정에서 배열의 모든 원소에서 **해당 원소를 끝으로 하는 최장 부분 증가 수열의 길이인 K[idx]**를 구할 수 있다.

카드 배열의 LIS와 LDS의 K를 구해서 더하면 아이디어 대로 비교할 수 있다. LDS는 배열을 뒤집어서 LIS를 구한 후 K를 뒤집은 것과 같다.

Idx	:	0	1	2	3	4	5	6
카드 숫자	:	2	1	4	2	3	6	3
K	:	0	0	1	1	2	3	2

idx:0	2					
Idx:1	1					
Idx:2	1	4				
Idx:3	1	2				
Idx:4	1	2	3			
Idx:5	1	2	3	6		
Idx:6	1	2	3	6		

이진 탐색을 이용한 LIS를 푸는 방법은 아래와 같다.

1. LIS의 길이를 나타내는 lis 배열을 하나 만들고, 첫 카드를 첫 번째 원소로 넣는다.
2. 다음 카드는 lis 배열에서 자신보다 같거나 첫 번째로 큰 위치에 들어간다. 없다면 배열의 마지막에 추가된다.
3. K 값은 카드가 배열에 들어가는 위치가 된다.
4. 모든 카드에 대해 반복한다.

2번에서 이진 탐색을 이용하기 때문에 O(logN)이 소요되고, 모든 카드에 대해 반복하기 때문에 전체 시간복잡도는 O(NlogN)이다.