

C6. Card Game

문제 분석

n을 카드의 숫자로 정의할 때, $1 \leq n \leq 50,000$ 개의 카드가 존재한다.
 k를 각 카드에 적혀있는 숫자로 정의할 때, $1 \leq k \leq 1,000,000,000$ 로 둘 수 있다. a 번째 카드의 k를 k_a 라고 정의한다.
 카드가 random하게 섞여있는 상태를 array로 보았을 때, 그 Longest Bitonic Subsequence를 구하라.

문제 풀이

문제 3개의 조건인 1) 카드가 하나도 없을 경우, 2) 전 카드보다 높은 카드를 선택해야하는 경우, 3) 전 카드보다 작은 카드를 선택할 경우, 다음부터는 무조건 전 카드보다 작은 카드를 선택해야 하는 경우를 한 단어로 표현하자면 Bitonic Sequence이다.
 Bitonic Subsequence란 $k_a < k_b < k_c \dots < k_g > k_i > k_j \dots$ (when a, b, c, ... g i j ... are in order)가 만족하는 sequence이다.
 Increasing Sequence와 Decreasing Sequence로 나누고, vector를 사용하는 방법으로 문제를 접근한다.

문제 풀이 분석

vector에 저장 되는 값: (현재 값보다 작은 카드들의 갯수(first), 해당 카드의 k(second))
 아래 두가지 규칙을 가지고 모든 card와 vector안의 값과 비교한다.

- k_x 가 현재까지 카드 중 가장 높은 숫자일 경우 -> (pair 중 가장 높은 second의 first+1, k_x) 추가
- k_x 가 현재까지 카드 중 가장 높은 숫자가 아닐 경우
 -> k_x 보다 작으면서 가장 큰 값을 가지는 second를 찾음
 -> 그 페어의 k_x 보다 작은 first+1한 pair의 second가 더 큰 경우 k_x 를 대체 (같은 first를 가진 값 중 second가 더 작은 값을 찾으면 대체)
 -> 위와 같은 방법 반복시 first들은 정렬된 형태를 띄게 되므로 같은 first값을 찾는 과정에서 binary search 사용
 위와 같은 방법을 increase에서 한 번 reverse해서 decrease에서 한 번 반복한다.

Time Complexity: $O(n \log(n))$
 [n개의 카드를 돌아보며 각각 binary search를 하기 때문]

Space Complexity: $O(n)$
 [n개의 카드 값]

	0	0,1	0,1,0	0,1,0,2	0,1,0,2,3	0,1,0,2,3,1	0,1,0,2,3,1,8	0,1,0,2,3,1,8,3	0,1,0,2,3,1,8,3,6	...
<-1,-1>	<-1,-1>	<-1,-1>	<-1,-1>	<-1,-1>	<-1,-1>	<-1,-1>	<-1,-1>	<-1,-1>	<-1,-1>	
	<0,0>	<0,0>	<0,0>	<0,0>	<0,0>	<0,0>	<0,0>	<0,0>	<0,0>	
		<1,1>	<1,1>	<1,1>	<1,1>	<1,1>	<1,1>	<1,1>	<1,1>	
				<2,2>	<2,2>	<2,2>	<2,2>	<2,2>	<2,2>	
					<3,3>	<3,3>	<3,3>	<3,3>	<3,3>	
							<4,8>	<4,8>	<4,6>	

Discussion

- 만약 같은 조건의 문제이지만 Longest Double Bitonic Sequence라면 문제를 어떻게 풀어야할까? 여기서 Double Bitonic Sequence란 Increasing→Decreasing→Increasing→Decreasing하는 Sequence를 뜻한다.