

고등부 2. 순서 섞기

풀이 작성자: 윤교준

배열 A 가 이미 비내림차순으로 정렬되어 있다면, 답은 0이다. 이제, A 가 비내림차순이 아니라고 가정한다.

부분문제 1 ($N \leq 8$)

길이 N 의 배열 A 에서 한 번의 순서 섞기 연산을 수행할 때, 나올 수 있는 결과의 종류의 수는 $O(2^N)$ 이다. 왜냐하면, 배열 A 의 좌측 혹은 우측 끝을 고르는 작업을 총 N 번 시행하기 때문이다.

‘순서 섞기’ 연산은 배열의 값의 순서만을 바꿀 뿐, 값을 바꾸지는 않는다. 고로, 배열 A 는 여러 번의 순서 섞기 연산을 통해, $O(N!)$ 종류의 값을 가질 수 있다.

따라서, BFS를 이용하여, 배열 A 가 정렬되기 위하여 필요한 연산의 최소 횟수를 $O(N! \times 2^N)$ 혹은 $O(N! \times N \times 2^N)$ 에 구할 수 있다.

부분문제 2 (답은 2 이하)

답이 0인 경우는, 배열 A 가 비내림차순인 경우이다. 이제, 답이 1인 경우의 필요충분조건을 알아보자.

답이 1이기 위한 필요조건

순서 섞기 연산을 f 라고 표현하자. 이제, 순서 섞기 연산의 역연산, 즉 f^{-1} 를 생각하자.

만일, A 에 f 를 취하여 B 를 얻었다고 하자. 이때, 배열 A 에서 ‘좌측의 수’로 선택된 수를 차례대로 배열 A_L 에 저장하고, A 에서 ‘우측의 수’로 선택된 수를 차례대로 배열 A_R 에 저장하자. A 에서, ‘좌측의 수’는 왼쪽부터 연속하게 붙어있고, ‘우측의 수’는 오른쪽부터 붙어있는 형태이다. 따라서, $A = A_L + A_R$ 임을 알 수 있다. 또한, 배열 B 는 두 배열 A_L, \widetilde{A}_R 을 “원소의 상대적인 순서를 유지하며” 합친 것과 같다. 여기서, 임의의 배열 T 에 대하여, \widetilde{T} 는 T 의 원소를 역순으로 배치한 배열을 의미한다.

따라서, 배열 B 에 대한 역연산 f^{-1} 는 다음과 같이 서술할 수 있다:

1. B 를 원소의 상대적인 순서를 유지하며, 두 개의 배열 B_L, B_R 로 나눈다.
2. $A = B_L + \widetilde{B}_R$

이제, 배열 A 가 단 한 번의 f 연산으로 정렬 가능하다고 가정하자. 이는, A 를 정렬한 배열 S 에 대하여, S 에 f^{-1} 연산을 취하여 A 를 얻을 수 있음을 의미한다. S 를 나눈 두 배열 S_L, S_R 은 모두 정렬되어 있다. $A = S_L + \widetilde{S}_R$ 이므로, A 는 증가하다가 감소하는 형태를 가져야 한다.

따라서, 답이 1이기 위한 필요조건은 “배열 A 가 증가하다가 감소하는 형태를 가지는 것”이다.

답이 1이기 위한 충분조건

이제, 배열 A 가 증가하다가 감소하는 형태를 가진다고 가정하자. 우리는 A 를 단 한 번의 순서 섞기 연산을 이용하여 정렬할 수 있음을 보일 것이다.

A 의 최댓값을 기준으로, (자신 포함) 왼쪽의 수를 모아둔 배열을 A_L , 오른쪽의 수를 모아둔 배열을 A_R 라고 하자. 가정에 의하여, A_L 과 \widetilde{A}_R 은 정렬된 상태이다.

A 에 연산 f 를 취한 결과 배열 B 는, A_L 과 \widetilde{A}_R 을 순서를 유지하며 합친 형태이다. 여기서, 합병 정렬 알고리즘의 아이디어를 이용하면, 정렬된 상태의 B 를 얻도록, 두 배열을 합치는 방법이 항상 존재함을 알 수 있다.

따라서, “증가하다가 감소하는 형태”를 가지는 배열은 항상 한 번의 연산으로 정렬할 수 있다.

위의 두 논의를 통하여, 답이 1인 경우의 필요충분조건은 “배열의 원소가 증가하다가 감소하는 형태를 가진다”임을 알 수 있다. 배열 A 가 이 조건을 만족하는지는 $O(N)$ 에 쉽게 확인할 수 있다. 답이 2인지는, 이 부분문제에 한하여, “답이 0과 1이 아니다”라는 조건으로 확인할 수 있다.

부분문제 3 ($1 \leq A_i \leq 2$)

배열 A 에서 연속한 1들과 연속한 2들을 각각 하나의 1과 2로 합쳐도 무방하다.

배열 A 에 K 개의 2가 존재한다고 하자. 배열 A 에서 $\lfloor \frac{K}{2} \rfloor$ 번째 2를 기준으로, 이 수까지를 A_L , 이 수의 오른쪽을 A_R 로 나누자. 두 배열 A_L, \widetilde{A}_R 를 합치는 적당한 방법이 존재하여, 배열 B 에 $\lfloor \frac{K}{2} \rfloor$ 개의 연속한 2가 존재하도록 할 수 있다.

즉, K 개의 ‘연속한 2’를, 한 번의 연산을 통해 $\lfloor \frac{K}{2} \rfloor$ 개의 ‘연속한 2’가 존재하도록 만들 수 있고, 이것이 최적의 전략이다.

따라서, 답은 $\lceil \log_2 K \rceil + 1$ 이다. 이에 대한 엄밀한 증명은 생략한다.

부분문제 4 (모든 A_i 가 서로 다르다)

부분문제 3의 풀이의 아이디어를 확장할 수 있다. 배열 A 가 총 K 번 ‘증가하다가 감소하는 형태’를 가진다면, 비슷한 논리로, 한 번의 연산을 이용하여, $\lfloor \frac{K}{2} \rfloor$ 번 ‘증가하다가 감소하는 형태’를 가지는 결과 배열을 얻을 수 있다.

이 전략 또한 최적이며, 답은 $\lceil \log_2 K \rceil + 1$ 이다.

부분문제 5

배열 A 에서 인접한 수가 서로 같다면, 이 두 수를 하나로 합쳐주어도 답이 변하지 않음을 알 수 있다. 이러한 성질을 이용하면, 부분문제 5 또한 부분문제 4와 동일한 풀이로 해결할 수 있다.

전체 시간 복잡도는 $O(N)$, 공간 복잡도는 $O(N)$ 혹은 $O(1)$ 에 해결할 수 있다.