균형잡힌 수열 – 풀이

작성자: 장근영

부분문제 1

아래 성질을 관찰할 수 있다.

- 균형잡힌 수열의 길이는 $2^n - 1$꼴로 나타낼 수 있다.
- 길이 $2^n - 1$의 균형잡힌 수열의 길이 $2^m - 1$의 접두사/접미사는 균형잡힌 수열이다. $(m \le n)$

높이 k인 균형잡힌 수열 내의 균형잡힌 부분수열의 개수를 $f(k)$로 정의하자. 전체 수열의 중앙값은 유일한 최댓값이므로, 이 값이 부분수열의 중심이 아닌 위치에 포함되면 최댓값 조건이 성립하지 않는다. 그리고 균형잡힌 수열의 접두사는 균형잡힌 수열이라는 관찰에 의해 중앙값을 중심으로 하는 k개의 부분수열은 균형잡힌 수열이다. 따라서 $f(k) = k + 2f(k-1)$을 만족한다. 점화식을 사용해 $O(\log N)$에 문제를 해결할 수 있다.

부분문제 2

각 원소의 크기가 3 이하인 경우에는 균형잡힌 수열 a을 아래와 같은 경우로 분류할 수 있다.

- a의 길이가 1
- $a_0 < a_1 > a_2$
- $a = [1, 2, 1, 3, 1, 2, 1]$

$O(N)$에 모든 경우를 셀 수 있다.

부분문제 3

원소를 변경하면서 생기거나 사라지는 균형잡힌 수열의 개수를 세주면 된다. 균형잡힌 수열의 길이는 최대 7이기 때문에 각 쿼리당 $O(1)$에 균형잡힌 수열 개수의 변화를 세줄 수 있다.

부분문제 4

길이가 $2^k - 1$ 형태인 모든 부분 구간 $A[i..j]$에 대하여 균형 잡힌 수열인지 여부를 판별하자. 판별 조건은 해당 구간의 중앙값이 구간 내 유일한 최댓값이어야 하며, 이를 기준으로 나뉜 좌우 부분 구간이 다시 재귀적으로 균형 잡힌 수열이어야 한다.

길이 $L = 2^k - 1$인 하나의 구간을 판별할 때, 최댓값 검증 및 재귀 호출을 수행하면 $O(L \log L)$의 연산이 필요하다. 이를 모든 구간에 대해 합산할 경우, 전체 시간 복잡도는 $O(N^2 \log N)$이 됩니다.

부분문제 5

쿼리의 개수가 적으므로, 변경 발생 시 전체 상태를 다시 계산하는 방식을 택한다. 이때 Dynamic Programming을 사용하여 작은 구간부터 상향식으로 계산하면 효율성을 높일 수 있다. 특히, 자식 구간이 균형 잡힌 수열임이 확인되었다면 자식 구간의 중앙값은 이미 해당 구간의 최댓값임이 보장된다. 따라서 전체를 순회할 필요 없이 현재 중앙값과 좌우 자식의 중앙값만을 비교하여 최댓값 조건을 $O(1)$에 검증할 수 있으며, 이를 통해 쿼리당 $O(N \log N)$에 해결 가능하다.

부분문제 6

고정된 k에 대해 특정 원소를 포함하는 길이 $2^k - 1$인 균형잡힌 수열은 최대 2개까지만 존재할 수 있다. 만약 3개 이상 존재한다고 가정하면, 그중 인접한 두 수열의 시작점 차이는 $2^{k-1} - 1$ 이하가 된다. 이는 필연적으로 한 수열이 다른 수열의 중앙값(해당 구간의 유일한 최댓값)을 자신의 구간 내 비중앙 위치에 포함하게 됨을 의미하며, 이는 최댓값 조건에 위배되어 모순이 발생한다.

이 관찰에 따라, 특정 원소의 값이 변경되었을 때 영향을 받아 재검사가 필요한 구간의 수는 각 높이마다 상수 개로 제한된다. 따라서 높이가 낮은 구간부터 상향식으로 올라가며 변경된 부분만을 갱신하면, 각 단계에서의 연산을 $O(1)$에 수행할 수 있습니다. 결과적으로 각 쿼리를 트리의 높이에 비례하는 시간 복잡도인 $O(\log N)$에 처리하여 전체 문제를 효율적으로 해결할 수 있습니다.