BOJ 15986
마법 목걸이

#greedy #segtree #two_pointer #sparse_table

난이도 – Diamond V

sorohue@sasa.hs.kr

BOJ 15986 마법 목걸이

- 이 문제를 푸는 방법은 다양합니다. 이 풀이보다 빠르고 간결한 풀이가 있을 수 있습니다.

- 세그먼트 트리 + 투 포인터 + 희소 배열을 이용하는 풀이입니다.

BOJ 15986 마법 목걸이

- 관찰을 통해, 구슬의 개수를 최대화하기 위해서는 구간을 가능한 한 작게 잡아 합치는 것이 항상 유리함을 알 수 있습니다.

- 목걸이를 잘랐을 때, 맨 왼쪽 구슬부터 순서대로 융합해주면서 각각 최종적으로 최대 몇 개의 구슬이 만들어지는 지 알아내면 됩니다.

BOJ 15986 **마법 목걸이**

- 목걸이의 1번 구슬부터 N번 구슬까지 나열한 것을
  그대로 뒤에 하나 더 붙여서 2N 크기의 배열을 만들어 줍시다.
- 그러면 첫 번째 구슬부터 N번째 구슬까지가 첫 구슬이 되도록
  잘랐을 경우를 배열 상의 길이가 N인 구간에 대응시킬 수 있습니다.

BOJ 15986 마법 목걸이

- d[i]를 i번째 구슬에서부터 구간을 잡았을 때
  융합되는 구슬들의 바로 오른쪽 구슬의 인덱스로 정의합시다.

  예를 들어, 인덱스는 1부터 시작한다고 했을 때
  [6, 12, 3, 5, 2, 7, 1] 라면 d[1] = 5입니다.
  구간을 [6, 12, 3, 5] 로 잡았을 때 처음으로 최대공약수가 1이 되고,
  따라서 그 다음 구슬의 인덱스인 5가 됩니다.

BOJ 15986 마법 목걸이

- d[i]가 2N을 초과하는 경우, d[i] = 2N+1로 정의합니다.

  d[2N+1] 또한 2N+1로 해 주면 이후 구현이 편리합니다.

- d[1]부터 d[2N]까지의 값을 모두 구했다면

  이제 문제는 1부터 N까지의 각 수에 대해 d[d[d[...[i]...]]] ≤ N+i 를

  만족시키면서 씌울 수 있는 d의 최대 개수를 구하는 것과 같습니다.

BOJ 15986 마법 목걸이

- 이 문제를 해결하기 위해 희소 배열을 사용할 수 있습니다.
- d[0][i] = d[i], d[k][i] = d[k−1][d[k−1][i]] 로 정의합니다.
- 그러면 d[k][i]는 i에 d를 2^k 번 씌웠을 때의 값이 됩니다.
- 마찬가지로, d[k][2N+1] = 2N+1로 정의합니다.

BOJ 15986 마법 목걸이

- 희소 배열을 구했다면 각각의 i에 대해 씌울 수 있는 d의 최대 개수는 O(lg N)에 할 수 있습니다.
- d를 씌울 수록 값이 순증가하기 때문에, k에 대해 역순으로 보면서 d[k][x] ≤ N+i 인 경우 x를 d[k][x]로 바꾸어 주고 답에 2^k를 더하면 됩니다.

BOJ 15986 마법 목걸이

- 희소 배열을 구하기 위해 세그먼트 트리와 투 포인터를 이용할 수 있습니다.
- 어떤 구슬에서 시작한 구간의 끝 구슬이 항상 그 왼쪽의 구슬들에서 시작한 구간의 끝 구슬보다 오른쪽에 있거나 같은 구슬임을 알 수 있습니다.
- 따라서, 세그먼트 트리 등에 구간 GCD를 저장해 두고 투 포인터를 진행하면서 왼쪽 포인터가 움직일 때만 새로 GCD를 구해 주면 O(N lg N)에 d[0][i]를 모두 구할 수 있습니다.