Approximating a Constant Range

Link submit: https://codeforces.com/contest/602/problem/B

Solution:

C++	https://ideone.com/Q3i5ft
Java	https://ideone.com/BJKZjV
Python	https://ideone.com/vY5o1E

Tóm tắt đề:

Trong giờ học, Xellos thực hành đo cường độ của một hiệu ứng đang dần tiến đến trạng thái cân bằng. Cách tốt để xác định cường độ cân bằng là chọn một số lượng lớn các điểm dữ liệu liên tục nhất có thể và lấy trung bình cộng.

Bạn được cho một dãy liên tục gồm n điểm dữ liệu $a_1, ..., a_n$. Biết rằng độ chênh lệch giữa hai điểm dữ liệu liên tục không vượt quá 1, tức $|a_{i+1} - a_i| \le 1$.

Một khoảng [l, r] của các điểm dữ liệu được cho là gần như cố định nếu như độ chênh lệch giữa giá trị lớn nhất M và nhỏ nhất trong dãy m không vượt quá 1, nghĩa là M - m ≤ 1. Tìm chiều dài lớn nhất của khoảng gần như cố định.

Input:

Dòng đầu tiên là một số nguyên *n* ($2 \le n \le 100.000$) – số lượng các điểm dữ liệu.

Dòng thứ hai gồm n số nguyên $a_1, a_2, ..., a_n$ ($1 \le a_i \le 100.000$).

Output:

In ra một số nguyên duy nhất là chiều dài lớn nhất của một khoảng gần như cố định.

Ví dụ:

5 1 2 3 3 2	4

11 5 4 5 5 6 7 8 8 8 7 6	5

Giải thích ví dụ:

Ví dụ 1: Khoảng gần như cố định dài nhất là khoảng [2, 3, 3, 2] có M = 3, m = 2 thỏa M - m ≤ 1 với độ dài là 4.

Ví dụ 2: Khoảng gần như cố định dài nhất là khoảng [7, 8, 8, 8, 7] có M = 8, m = 7 thỏa M - m ≤ 1 với độ dài là 5.

Hướng dẫn giải:

Nhận xét:

- Cần chú ý rằng độ chênh lệnh giữa hai điểm dữ liệu liên tục không vượt quá 1. Do đó một đoạn hợp lệ là đoạn liên tục có số lượng phần tử phân biệt không vượt quá 2.
- Để lần lượt duyệt qua từng đoạn liên tục, ta sử dụng kỹ thuật Two Pointers với hai biến chạy i và j. Ban đầu ta sẽ mở rộng đoạn bằng cách sử dụng biến i cho tới khi nào đoạn của ta có số lượng phần tử phân biệt lớn hơn 2, lúc này ta tiến hành rút ngắn đoạn bằng cách đưa j chạy ngược lên. Thực hiện tương tự cho đến hết mảng.
- Sử dụng mảng đếm phân phối để nhanh chóng tìm được số lượng phần tử phân biệt của đoạn đang xét.

Như vậy, ta có cách giải quyết bài toán này như sau:

- Bước 1: Đưa thông tin các điểm dữ liệu vào mảng.
- Bước 2: Khởi tạo biến lưu số lượng phần tử phân biệt của đoạn hiện có và mảng đếm phân phối.
- Bước 3: Khởi tạo biến chạy j ở đầu mảng.
- Bước 4: Cho biến i chạy từ đầu mảng:
    - Nếu phần tử hiện tại ở vị trí i lần đầu tiên xuất hiện (giá trị trong mảng phân phối bằng 0) thì tăng biến đếm số lượng phân biệt lên 1.
    - Nếu số lượng phần tử phân biệt lúc này lớn hơn 2, thực hiện rút ngắn đoạn bằng biến j. Đồng thời cập nhật số lượng phần tử phân biệt.
    - So sánh đoạn hiện tại [j, i] có kích thước i - j + 1 với độ dài lớn nhất hiện có và tiến hành cập nhật.
- Bước 5: In ra kết quả.

Độ phức tạp: **O(n)** với n là số điểm dữ liệu.