

Số khác nhau

Giới hạn thời gian: 1.0s Giới hạn bộ nhớ: 256M

Bài toán truy vấn trên dãy: với mỗi đoạn con $A[L..R]$, hãy đếm số lượng phần tử **phân biệt** đang xuất hiện trong đoạn đó. Cần xử lý nhiều truy vấn hiệu quả với ràng buộc lớn.

Yêu cầu

Cho dãy số nguyên dương A độ dài N . Với mỗi truy vấn $[L,R]$, hãy tính số lượng giá trị khác nhau xuất hiện trong đoạn $A[L..R]$.

Dữ liệu

- Dòng đầu chứa hai số nguyên dương $N, Q \sim (1 \leq N, Q \leq 10^5)$.
- Dòng thứ hai chứa N số nguyên dương biểu diễn dãy A với $1 \leq A_i \leq 10^6$.
- Mỗi trong Q dòng tiếp theo chứa hai số nguyên $L, R \sim (1 \leq L \leq R \leq N)$ mô tả một truy vấn.

Kết quả

- Gồm Q dòng, dòng thứ i là số lượng phần tử phân biệt trong $A[L_i..R_i]$.

Ví dụ

Ví dụ 1

Input

```
6 2
1 4 5 3 5 2
1 3
3 5
```

Output

```
3
2
```

Ví dụ 2

Input

```
5 3
1 1 2 1 2
1 5
2 4
3 3
```

Output

```
2
2
1
```

Giải thích

Ví dụ 1

- Với $[1,3]$, đoạn $A[1..3]=[1,4,5]$ có 3 giá trị phân biệt $\{1,4,5\}$.
- Với $[3,5]$, đoạn $A[3..5]=[5,3,5]$ có 2 giá trị phân biệt $\{3,5\}$.

Ví dụ 2

- $[1,5]$: $[1,1,2,1,2]$ có 2 giá trị phân biệt $\{1,2\}$.
- $[2,4]$: $[1,2,1]$ có 2 giá trị phân biệt $\{1,2\}$.
- $[3,3]$: chỉ có $[2]$, nên kết quả 1.

Chăm điểm

- 100% số điểm cho lời giải $\mathcal{O}((N+Q)\log N)$ hoặc $\mathcal{O}((N+Q)\sqrt{N})$, ví dụ:
 - Cách 1 (khuyến nghị): xử lý offline theo R tăng dần, dùng [Fenwick](#) / [Segment Tree](#) với 1 vị trí xuất hiện cuối cùng:
 - Duyệt $R=1$ to N , tại mỗi bước: nếu giá trị A_R đã xuất hiện ở vị trí cũ p , giảm đóng góp tại p , rồi tăng tại R .
 - Trả lời các truy vấn kết thúc tại R bằng tổng trên $[L,R]$ (đếm số vị trí **đại diện** đang bật).
 - Cách 2: [Mo's algorithm](#) sắp xếp truy vấn theo khối $\sqrt{N}$, duy trì đếm tần suất và số giá trị đang có tần suất >0 .
- Dùng nhập/xuất nhanh cho $N, Q \leq 10^5$. Có thể nén giá trị nếu miền A_i lớn.