

Sắp xếp ảo

Đề bài

Cho một mảng gồm các số nguyên không âm.

Nhận được các truy vấn dưới dạng "gán một giá trị nào đó cho một phần tử". Sau mỗi truy vấn, cần phải xuất ra số nguyên không âm nhỏ nhất x (nếu nó tồn tại) sao cho mảng $a \oplus x$ được sắp xếp.

Giải pháp 30 điểm $O(qnA)$

Lưu ý rằng nếu tất cả các số đều nhỏ hơn 2^k , thì x nhỏ nhất cũng sẽ nhỏ hơn 2^k , bởi vì các bit cao hơn sẽ không ảnh hưởng đến việc so sánh. Chúng ta thực hiện như sau:

Sau mỗi truy vấn, thử các giá trị của x .

Với mỗi $1 \leq i \leq n-1$, kiểm tra rằng $a_i \oplus x \leq a_{i+1} \oplus x$.

Nghiên cứu

Mảng được sắp xếp nếu đối với mỗi cặp phần tử liền kề, điều kiện $a_i \leq a_{i+1}$ được thỏa mãn.

Hãy xem xét cách các số a_i và a_{i+1} ảnh hưởng đến x .

Nếu các số bằng nhau, chúng không đặt ra bất kỳ giới hạn nào.

Nếu không, bit cao nhất mà tại đó các số khác nhau sẽ đặt ra giới hạn cho x .

Như vậy, có $n-1$ giới hạn đối với x dưới dạng "bit thứ i bằng 0 hoặc 1".

Giải pháp 59 điểm $O(qnlogA)$

Sau mỗi truy vấn, đối với mỗi $1 \leq i \leq n-1$, chúng ta sẽ xác định giới hạn cho x . Nếu các giới hạn mâu thuẫn (tức là đồng thời có $\{i, 0\}$ và $\{i, 1\}$, thì không tồn tại đáp án).

Ngược lại, đối với mỗi giới hạn $\{i, 1\}$, chúng ta đặt bit thứ i trong x .

Dễ dàng chứng minh rằng x như vậy là nhỏ nhất.

Giải pháp 80 điểm $O(q \log n \log A)$

Sử dụng cấu trúc dữ liệu:

Chúng ta sẽ lưu trữ các giới hạn trong một cây phân đoạn (segment tree).

Ở mỗi đỉnh, đối với mỗi cặp $\{i, 0/1\}$ U lưu trữ 0 hoặc 1 để chỉ ra liệu có tồn tại giới hạn này trên đoạn đó hay không.

Truy vấn thay đổi được thực hiện bằng cách đi lên cây trong $O(\log n \log A)$.

Trả lời truy vấn được hình thành tại gốc của cây phân đoạn trong $O(\log A)$.

Giải pháp hoàn chỉnh $O(q \log n + \log A)$

Tối ưu hóa cây phân đoạn:

Lưu ý rằng trong giải pháp trước, mỗi đỉnh của cây lưu trữ 60 giá trị boolean. Chúng ta sẽ sử dụng nén bit.

Truy vấn thay đổi thực hiện bằng cách đi lên cây trong $O(\log n)$.

Trả lời truy vấn được hình thành tại gốc của cây phân đoạn trong $O(\log A)$.

Giải pháp hoàn chỉnh $O(q \log A)$

Loại bỏ cây phân đoạn:

Lưu ý rằng chúng ta chỉ sử dụng gốc của cây phân đoạn để trả lời truy vấn.

Thay thế cây đoạn bằng một mảng (số lượng giới hạn cho mỗi cặp $\{i, 0/1\}$).

Xử lý truy vấn và trả lời trong $O(\log A)$.

Các tối ưu hóa tiếp theo

Làm việc với các bit:

Điều gì làm chậm chúng ta?

Tìm bit cao nhất trong số x .

Sử dụng hàm `__builtin_clz` (đếm số bit 0 đứng đầu trong $O(\log w)$).

Trên một số máy tính, có sẵn lệnh hợp ngữ BSR.

Tính trước câu trả lời cho tất cả các từ 16 bit.