

Bài A. ANDGAME

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 1 giây

Trò chơi AND được mô tả như sau:

- Cho mảng A có n số nguyên: a_1, a_2, \dots, a_n , mỗi số đều thuộc $[0, 2^{63} - 1]$
- Có 2 người chơi, luân phiên nhau thực hiện nước đi
- Đến lượt mình, người chơi chọn một số a_i tùy ý trong mảng A , và một số nguyên x tùy ý thuộc $[0, 2^{63} - 1]$ sao cho $a_i \text{ AND } x \neq a_i$. Sau đó thay a_i bởi $a_i \text{ AND } x$. Đây gọi là một nước đi hợp lệ. Ở đây AND là phép toán và trên từng bit của hai số nguyên không âm (tức &)
- Ai không thực hiện được nước đi hợp lệ nữa sẽ thua cuộc. Rõ ràng là trò chơi sẽ kết thúc sau hữu hạn bước, nên sẽ không có kết quả hòa

Biết rằng 2 người chơi đều rất thông minh, hãy tính số cách thực hiện nước đi đầu tiên, sao cho người đi trước luôn thắng

Dữ liệu vào

- Dòng đầu tiên chứa n
- Dòng tiếp theo chứa dãy A

Kết quả

In ra số cách theo mod 1000000007

Ví dụ

stdin	stdout
4 5 5 9	0

Hạn chế

- $1 \leq n \leq 10^5$
- 30% test có $n = 2$

Bài B. LNET

File dữ liệu vào: **stdin**
File kết quả: **stdout**
Hạn chế thời gian: 1 giây

Hệ thống mạng trên hành tinh XYZ thỏa mãn tính chất sau: Giữa hai đỉnh x, y bất kỳ, tồn tại và duy nhất một đường đi đơn giữa chúng và được ký hiệu là $\text{Path}(x, y)$. Nói cách khác, hệ thống mạng có dạng hình cây. Có một số cặp nút mạng đang truyền thông tin cho nhau, gọi là các kết nối. Với a là một kết nối, ký hiệu s_a và t_a lần lượt là đỉnh gửi và đỉnh nhận ($s_a \neq t_a$).

Ta nói kết nối a và kết nối b là xung đột nhau nếu chúng dùng chung ít nhất một nút mạng, tức là $\text{Path}(s_a, t_a)$ và $\text{Path}(s_b, t_b)$ có đỉnh chung. Lúc này, cần phải dùng hai cách mã hóa khác nhau để mã hóa cho đường truyền a và b . Câu hỏi đặt ra là: Cần dùng ít nhất bao nhiêu cách mã hóa khác nhau để gán cho mỗi kết nối một cách mã hóa, đảm bảo hai kết nối xung đột nhau thì dùng hai cách mã hóa khác nhau.

Các kết nối trên mạng có tính trực tuyến. Ban đầu chưa có kết nối nào, sau đó có thể có thêm các kết nối hoặc một số kết nối mất đi. Sau mỗi lần biến đổi như vậy, hãy tính toán và đưa ra số cách mã hóa ít nhất cần dùng.

Dữ liệu vào

- Dòng đầu chứa hai số nguyên dương: n m là số đỉnh của cây và số thay đổi của mạng
- n dòng tiếp theo mỗi dòng ghi một cạnh của cây: u v
- m dòng tiếp theo mỗi dòng ghi một biến đổi của mạng: k s t với s là đỉnh gửi, t là đỉnh nhận, $k = 0/1$ tương ứng là có thêm hoặc mất đi một kết nối từ s đến t

Dữ liệu đảm bảo có ít nhất một kết nối từ s đến t khi $k = 1$, và nếu có nhiều kết nối từ s đến t thì mỗi lần chỉ mất đi một trong số đó. Các đỉnh của cây được đánh số từ 1.

Kết quả

- Ghi m dòng là kết quả tính được sau mỗi biến đổi

Ví dụ

stdin	stdout
7 5	1
1 2	2
1 3	3
2 4	4
2 5	3
3 6	
3 7	
0 2 7	
0 3 1	
0 1 4	
0 4 6	
1 2 7	

Hạn chế

- $1 \leq n, m \leq 10^5$
- Subtask 1: $1 \leq n, m \leq 1000$
- Subtask 2: $1 \leq m \leq 1000$
- Subtask 3: $1 \leq n \leq 1000$
- Subtask 4: Ràng buộc gốc

Bài C. HALFGAME

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 1 giây

An và Bình chơi một trò chơi như sau:

- Có n đồng sỏi, đồng thứ i có a_i viên sỏi
- Mỗi lượt chơi, người chơi chọn ra một đồng sỏi và lấy đi một ít sỏi ở đó, ít nhất là một viên và nhiều nhất là $(k + 1)/2$ viên, với k là số sỏi hiện tại ở trong đồng được chọn
- Hai người luân phiên nhau thực hiện lượt chơi, An chơi trước. Ai không thực hiện được lượt chơi nữa thì thua cuộc

Xác định xem liệu An có thể chiến thắng trò chơi hay không, nếu như Bình rất thông minh

Dữ liệu vào

Gồm 5 test case, mỗi test case trên một dòng:

- Số đầu tiên là n
- Tiếp theo là n số: a_1, a_2, \dots, a_n

Kết quả

Gồm 5 dòng, mỗi dòng ghi yes/no tương ứng là An có thể chiến thắng hoặc không

Ví dụ

stdin	stdout
1 14	no
1 10	yes
2 14 15	yes
2 20 20	no
3 1 2 3	yes

Hạn chế

- $1 \leq n \leq 10^5$
- $1 \leq a_i \leq 10^{18}$
- Subtask 0: $n = 1$ và $a_i \leq 1000$
- Subtask 1: $n = 1$
- Subtask 2: $n = 3$ và $a_i \leq 100$
- Subtask 3: $a_i \leq 1000$
- Subtask 4: Ràng buộc gốc

Bài D. ARRCNT

File dữ liệu vào: **stdin**
File kết quả: **stdout**
Hạn chế thời gian: 1 giây

Một dãy số nguyên dương $a = a_1, a_2, \dots, a_n$ được gọi là mượt mà nếu $|a_{i+1} - a_i| \leq 1 \forall i \in \{1, 2, \dots, n-1\}$. Huyền đang có một dãy số nguyên dương mượt mà, nhưng cô đã quên mất một vài số trong dãy đó.

Huyền muốn khôi phục lại dãy a bằng cách thay những số không nhớ bằng một số nguyên dương nào đó, sao cho dãy thu được vẫn mượt mà. Cô nhận ra có thể có rất nhiều dãy thỏa mãn. Trong phút yếu lòng, cô muốn nhờ sự giúp đỡ của bạn. Hãy giúp cô ấy tính xem có bao nhiêu dãy khác nhau có thể tạo ra.

Dữ liệu vào

- Dòng đầu tiên chứa số nguyên dương n
- Dòng thứ hai chứa $a_1 \ a_2 \ \dots \ a_n$. Những số bị quên được biểu thị bằng số 0, dữ liệu đảm bảo có ít nhất một số khác 0 trong dãy.

Kết quả

Ghi số dãy khác nhau có thể tạo ra sau khi chia lấy dư cho $10^9 + 7$

Ví dụ

stdin	stdout
4 2 0 1 0	4
6 7 0 0 0 0 1	0

Hạn chế

- $1 \leq n \leq 10^6$, $0 \leq a_i \leq 10^9$ trong tất cả các test.
- 20% số test với $n \leq 1000$.
- 20% số test với các số Huyền nhớ đều lớn hơn hoặc bằng n .
- 20% số test với các số Huyền nhớ đều bằng 1 và Huyền nhớ a_1, a_n .
- 20% số test Huyền nhớ a_1, a_n
- 20% test còn lại không có ràng buộc gì thêm.