

## Bài A. RGAME

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

Các học sinh trong lúc rảnh rỗi đã nghĩ ra một trò chơi như sau:

- Có  $n$  viên bi trên bàn, hai người chơi sẽ luân phiên nhau lấy
- Đến lượt mình, người chơi lấy đi một số viên bi từ số bi trên bàn, ít nhất là một viên và nhiều nhất là toàn bộ
- Lượt chơi đầu tiên phải lấy ít hơn  $n$  viên. Các lượt sau đó phải lấy không quá số bi vừa được lấy bởi đối thủ
- Ai không thể thực hiện được lượt chơi của mình thì thua cuộc

Bạn được cho trước  $n$  và quyền chọn lượt chơi, hãy giành chiến thắng trong trò chơi này!

Tương tác:

Đầu tiên bạn đọc vào số  $n$  từ đầu vào chuẩn. Sau đó là các lượt chơi của hai bên được thực hiện, cho đến khi một trong hai bên thắng cuộc thì chương trình sẽ tự động ngắt. Ở mỗi lượt chơi:

- Bạn cần in ra đầu ra chuẩn một số nguyên  $x$  để thực hiện lấy đi  $x$  viên trên bàn ở lượt chơi của bạn. Ở lượt chơi đầu tiên bạn có thể để  $x = 0$  nếu muốn máy chơi trước.
- Đọc vào số  $x$  từ đầu vào chuẩn,  $x$  là số viên mà máy chơi ở lượt của máy.

Lưu ý bạn cần flush stdout (hoặc endl) sau mỗi dòng in ra.

### Ví dụ

stdin	stdout
5	0
2	1
1	1

### Hạn chế

- $n \leq 10^6$
- Có 50% số test với  $n \leq 1000$

## Bài B. ZGAME

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

Cho  $n$  con rùa xếp thành một hàng, đánh số từ 1 đến  $n$  từ trái sang phải. Mỗi con rùa ở một trong hai trạng thái ngửa hoặc sấp. Xét một trò chơi như sau:

- Có hai người, luân phiên nhau thực hiện lượt chơi
- Đến lượt mình, người chơi chọn tùy ý một con rùa đang sấp và lật ngửa nó, đồng thời có thể lựa chọn lật ngược hoặc không lật ngược con rùa ngay bên trái nó (nếu có). Lật ngược có nghĩa là chuyển sấp thành ngửa và ngửa thành sấp
- Ai không thực hiện được lượt chơi 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

Bạn sẽ chơi trò này với máy, bạn được quyền chọn người đi trước và hãy dành chiến thắng trong trò chơi.

Tương tác:

- Đầu tiên bạn cần đọc vào số  $n$  ( $1 \leq n \leq 5000$ )
- Tiếp theo bạn cần đọc vào một xâu nhị phân độ dài  $n$ , bit thứ  $i$  là 1/0 cho biết con rùa đang sấp/ngửa
- Sau đó bạn cần in ra 1 hoặc 0 tương ứng là bạn muốn đi trước hoặc đi sau
- Sau đó trò chơi sẽ bắt đầu. Khi đến lượt máy, máy sẽ in ra một số mô tả lượt chơi mà máy thực hiện, bạn cần phải đọc vào số này và chuyển sang lượt chơi của bạn. Khi đến lượt bạn, bạn cần in ra một số mô tả lượt chơi mà bạn thực hiện và chuyển sang lượt chơi của máy
- Số dùng để mô tả một lượt chơi là số  $i$  ( $-n \leq i \leq n, i \neq 0, i \neq -1$ ) với ý nghĩa: Nếu  $i > 0$  thì lượt chơi này sẽ lật ngược con rùa thứ  $i$ , nếu  $i < 0$  thì lượt chơi này sẽ lật ngược con rùa thứ  $-i$  và  $-i - 1$
- Trò chơi sẽ kết thúc khi tất cả rùa đều ngửa

Lưu ý, sau mỗi lần in ra bạn cần đẩy dữ liệu ra luồng chuẩn (`flush(stdout)` hoặc `cout « endl`) để tương tác được với máy.

### Ví dụ

stdin	stdout
3	1
111	2
-3	-2

## Bài C. 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 9	0

### Hạn chế

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

## Bài D. EZROOKGAME

File dữ liệu vào: `stdin`  
File kết quả: `stdout`  
Hạn chế thời gian: 2 giây

Trò chơi ROOK được mô tả như sau:

- Cho bàn cờ  $n$  dòng  $m$  cột, trên đó có  $k$  quân xe (không có màu). Quân thứ  $i$  ở dòng  $x_i$  và cột  $y_i$ . Bàn cờ này hơi khác so với thực tế, mỗi ô có thể chứa được nhiều quân cờ cùng lúc, và luật di chuyển của xe trong trường hợp này là: Chỉ được phép di chuyển lên phía trên (cùng cột) hoặc sang bên trái (cùng dòng), ít nhất là một bước và nhiều nhất là đi đến cạnh của bàn cờ
- Có hai người chơi, luân phiên nhau thực hiện nước đi
- Đến lượt mình, người chơi chọn một quân xe tùy ý trên bàn và di chuyển nó sang một ô khác (theo luật di chuyển ở trên)
- 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 xác định xem liệu người đi trước có thắng hay không

### Dữ liệu vào

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

- Dòng đầu tiên chứa  $n, m, k$
- Dòng tiếp theo chứa  $x_1, y_1, x_2, y_2, \dots, x_k, y_k$

### Kết quả

Gồm 5 dòng, ghi YES/NO tương ứng là người đi trước thắng/thua

### Ví dụ

stdin	stdout
8 8 3	YES
1 1 1 2 1 3	NO
8 8 3	NO
1 2 1 3 1 4	NO
100 100 1	NO
100 100	
10 10 2	
8 10 10 8	
10 10 2	
9 9 10 10	

### Hạn chế

- $1 \leq n, m \leq 10^{18}, 1 \leq k \leq 10^5$
- Subtask 0:  $n = 1$
- Subtask 1:  $n, m \leq 100$
- Subtask 2:  $n, m \leq 1000$
- Subtask 3: Ràng buộc gốc

## Bài E. BROADGAME

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

Tý và Tí là đôi bạn thân cùng đội tuyển. Một hôm họ rảnh rỗi đã nghĩ ra một trò chơi đối kháng như sau:

- Cho  $k$  bảng 2 chiều chứa các phân số, bảng thứ  $i$  có kích thước  $n_i \times m_i$ . Ô có tọa độ  $(i, j)$  sẽ ghi phân số  $\frac{i}{j}$
- Hai người luân phiên thực hiện lượt chơi, Tý đi trước. Đến lượt mình, người chơi chọn ra một bảng trong số  $k$  bảng trên, chọn một ô trên bảng và tô màu nó, sao cho ô được chọn có giá trị lớn hơn mọi ô khác đã được tô màu trên bảng đó
- Đến lượt mình, ai không thực hiện được lượt chơi nữa thì thua. Cả hai người đều rất thông minh - có lẽ vì học đội tuyển chẳng

Hãy giúp Tý xác định rằng mình có thể thắng trong trò chơi này hay không

### Dữ liệu vào

Gồm 3 bộ dữ liệu trong 1 test, mỗi bộ bao gồm:

- Dòng đầu:  $k$
- $k$  dòng tiếp theo, dòng thứ  $i$  ghi:  $n_i \ m_i$

### Kết quả

- Gồm 3 dòng ghi YES/NO tương ứng là Tý có/không thể thắng

### Ví dụ

stdin	stdout
5	NO
1 1	YES
1 2	NO
2 3	
3 4	
4 5	
3	
7 7	
8 8	
9 9	
4	
2 2	
3 3	
4 4	
4 5	

### Hạn chế

- Subtask 0:  $1 \leq n, m, k \leq 200$
- Subtask 1:  $1 \leq n, m \leq 2000, k \leq 20000$
- Subtask 2:  $1 \leq n, m, k \leq 20000, n = m$
- Subtask 3:  $1 \leq n, m, k \leq 20000$

## Bài F. TGAMEZ

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

Cây là một đồ thị vô hướng liên thông không có chu trình. Cho một cây có  $n$  đỉnh, các đỉnh được đánh số từ 1 đến  $n$  với gốc là đỉnh 1. Nút thứ  $i$  của cây được đặt  $a_i$  đồng xu

Xét một trò chơi thú vị trên cây này như sau:

- Có hai người chơi. Hai người chơi này sẽ luân phiên thực hiện lượt chơi của mình
- Mỗi lượt chơi, người chơi chọn ra một nút không phải là nút gốc của cây và chuyển một số đồng xu ở đó lên nút cha của nó (ít nhất là một và nhiều nhất là toàn bộ)
- Ai không thực hiện được lượt chơi nữa thì thua cuộc

Hãy tính xem người đi trước có bao nhiêu cách thực hiện nước đi đầu tiên sao cho chắc chắn dành được chiến thắng. Hai cách đi được coi là khác nhau nếu nút được chọn khác nhau hoặc số lượng đồng xu được chuyển đi khác nhau

### Dữ liệu vào

- Dòng đầu chứa số nguyên dương:  $n$
- $n - 1$  dòng tiếp theo, dòng thứ  $i$  chứa hai số nguyên không âm là đỉnh cha của đỉnh  $i + 1$  và  $a_{i+1}$

$a_1$  không ảnh hưởng đến trò chơi nên không cần nhập vào

### Kết quả

Ghi số cách tìm được

### Ví dụ

stdin	stdout
4 1 2 1 4 1 5	1

### Hạn chế

- $2 \leq n \leq 10^5$ ,  $0 \leq a_i \leq 10^9$
- Subtask 1: Độ sâu của cây (số cạnh nhiều nhất trên đường đi từ gốc đến nút lá) bằng 1
- Subtask 2: Độ sâu của cây (số cạnh nhiều nhất trên đường đi từ gốc đến nút lá) bằng  $n - 1$
- Subtask 3: Ràng buộc gốc