

Dãy số

Alice có một số nguyên dương M và dãy số nguyên không âm a_1, a_2, \dots, a_n , trong đó a_i có giá trị bằng 0 hoặc 1. Alice cần thực hiện q thao tác thuộc một trong hai loại sau:

- Thao tác loại 1 có dạng: $1 L R c$ với $1 \leq L \leq R \leq n$ và $0 < c \leq 10^9$, thao tác này thực hiện thay các phần tử a_i bằng $(a_i + c) \% M$ với $L \leq i \leq R$. Chú ý, phép toán $\%$ là phép toán chia lấy dư;
- Thao tác loại 2 có dạng: $2 L R$ với $1 \leq L < R \leq n$, kiểm tra xem trong đoạn $[L, R]$ có ít nhất hai phần tử khác nhau không? Cụ thể, thao tác này đưa ra 1 nếu tồn tại hai chỉ số i, j mà $L \leq i < j \leq R$ và $a_i \neq a_j$, ngược lại đưa ra giá trị 0.

Yêu cầu: Giúp Alice thực hiện lần lượt q thao tác, với mỗi thao tác loại 2 ghi ra câu trả lời.

Dữ liệu:

- Dòng đầu tiên chứa ba số nguyên dương n, M, q ($n, q \leq 2 \times 10^5; M \leq 10^9$);
- Dòng thứ hai chứa n số nguyên không âm a_1, a_2, \dots, a_n ($0 \leq a_i \leq 1$);
- Dòng thứ k ($1 \leq k \leq q$) trong q dòng tiếp theo chứa các số nguyên mô tả thao tác thứ k .

Kết quả: Ghi một số dòng, mỗi dòng tương ứng là câu trả lời cho thao tác loại 2, lần lượt tương ứng trong dữ liệu vào.

Ràng buộc:

- Subtask 1 (20% số điểm): $n, q \leq 10^3; M = 2$ và các thao tác loại 1 đều có $L = R$;
- Subtask 2 (20% số điểm): Không có thao tác loại 1;
- Subtask 3 (20% số điểm): $M = 2$ và các thao tác loại 1 đều có $L = R$;
- Subtask 4 (20% số điểm): $M = 2$;
- Subtask 5 (20% số điểm): Không có ràng buộc nào thêm.

Input	Output
4 2 4	1
1 0 1 0	0
2 1 2	1
1 2 2 1	
2 1 3	
2 1 4	

Thả trứng (egg)

Bạn có e quả trứng giống hệt nhau và một tòa nhà gồm n tầng. Bạn cần xác định số lần thả ít nhất để xác định được tầng cao nhất mà khi thả quả trứng từ tầng đó xuống trứng vẫn không vỡ hoặc trứng sẽ vỡ khi thả từ tầng 1. Một quả trứng thả xong mà không bị vỡ bạn có thể dùng lại.

Input

- Gồm một dòng chứa hai số nguyên e, n ($n \leq 10^6$).

Output

- Gồm một dòng chứa một số là số lần thả ít nhất.

Input	Output
1 5	5
2 5	3

Subtask 1: $e \leq 2; n \leq 1000$;

Subtask 2: $e \leq 10; n \leq 1000$;

Subtask 3: $e \leq 100$;

Subtask 4: $e \leq 1000$;

Thả trứng tương tác với hệ thống (eggz)

Bạn có e quả trứng giống hệt nhau và một tòa nhà gồm n tầng.

Bạn cần xây dựng hàm `int solve(int E, int N)` để xác định giá trị t lớn nhất (là giá trị mà khi thả quả trứng từ tầng t xuống trứng vẫn không vỡ) với số lần thả ít nhất.

Mỗi lần thả bằng cách gọi hàm `bool drop(int f)`

Chú ý: Một quả trứng thả xong mà không bị vỡ bạn có thể dùng lại. Chương trình cần `#include "egg.h"`

Cài đặt

```
#include <bits/stdc++.h>
#include "egg.h"

// Xây dựng hàm solve xác định tầng t lớn nhất
int solve(int E, int N);

// Hàm solve có thể gọi hàm drop(f), hàm sẽ trả về trứng bị vỡ
// (false) hay không vỡ (true) khi thả từ tầng f
bool drop(int f);
```

Tối ưu biến đổi

Alice có một bức ảnh đen trắng được biểu diễn bằng một mảng P kích thước $m \times n$, mỗi phần tử chỉ nhận một trong hai giá trị 0/1 (tương ứng điểm ảnh màu trắng hoặc màu đen). Xét các thao tác biến đổi trên bức ảnh biểu diễn bằng mảng P , giả sử mảng Q là mảng kết quả.

- Biến đổi 1: $Q_{j,i} = P_{i,j}$
- Biến đổi 2: $Q_{n-j+1,m-i+1} = P_{i,j}$
- Biến đổi H : $Q_{m-i+1,j} = P_{i,j}$
- Biến đổi V : $Q_{i,n-j+1} = P_{i,j}$
- Biến đổi quay 90° (kí hiệu A), 180° (kí hiệu B), 270° (kí hiệu C) theo chiều kim đồng hồ, ví dụ, biến đổi A : $Q_{j,m-i+1} = P_{i,j}$
- Biến đổi quay 90° (kí hiệu X), 180° (kí hiệu Y), 270° (kí hiệu Z) ngược chiều kim đồng hồ, ví dụ, biến đổi X : $Q_{n-j+1,i} = P_{i,j}$

Yêu cầu: Cho mảng P và một dãy phép biến đổi S , hãy tìm một dãy biến đổi tối ưu với ít phép biến đổi nhất mà kết quả biến đổi giống như thực hiện với dãy biến đổi S .

Dữ liệu:

- Dòng đầu gồm hai số m, n ($1 \leq m, n \leq 1024$);
- m dòng sau, mỗi dòng n số mô tả mảng P ;
- Dòng tiếp theo là một xâu S (độ dài không vượt quá 10^5) chỉ gồm các kí tự '1', '2', 'H', 'V', 'A', 'B', 'C', 'X', 'Y', 'Z'.

Kết quả: Ghi ra một dòng chứa một xâu mô tả dãy biến đổi tối ưu tìm được, xâu biến đổi được đặt trong cặp ngoặc vuông.

Ví dụ:

Input	Output
2 3 0 0 0 0 0 1 BC	[A]
2 2 0 0 0 0 12HVABCXYZ	[]

Ràng buộc:

- Có 30% số test có $m = n \leq 4$;
- Có 30% số test khác có độ dài xâu S không vượt quá 100;
- Có 40% số test còn lại không có ràng buộc nào thêm.

mex

Xét một bảng số kích thước $n \times n$, kí hiệu phần tử $a_{i,j}$ là giá trị ở hàng i ($1 \leq i \leq n$) và cột j ($1 \leq j \leq n$). Bảng đã được điền giá trị ở hàng 1 và cột 1 bằng các số 0, 1, 2, các phần tử còn lại $a_{ij} = \text{mex}(a_{i-1,j}, a_{i,j-1})$, trong đó mex được xác theo bảng sau:

mex(x, y)	y = 0	y = 1	y = 2
x = 0	1	2	1
x = 1	2	0	0
x = 2	1	0	0

Yêu cầu: Đếm số phần tử của bảng nhận giá trị bằng 0, 1, 2?

Input

- Dòng đầu chứa số nguyên dương n ($n \leq 5e5$);
- Dòng thứ hai chứa n số $a_{1,1}, a_{1,2}, \dots, a_{1,n}$ là các số đã được điền ở hàng 1;
- Dòng thứ ba chứa $n - 1$ số $a_{2,1}, a_{2,2}, \dots, a_{n,1}$ là các số được điền ở cột 1.

Output

- Gồm ba số tương ứng là số phần tử của bảng nhận giá trị bằng 0, 1, 2.

Input	Output
<pre>3 0 1 2 1 2</pre>	<pre>3 4 2</pre>

book1

Alice có một quyển sách gồm n chương, chương thứ i ($1 \leq i \leq n$) cần đọc mất t_i đơn vị thời gian. Alice dự định đọc quyển sách làm k đợt, mỗi đợt sẽ đọc một số chương liên tiếp nhau để thời gian đợt đọc nhiều nhất là nhỏ nhất. Cụ thể, Alice cần chia dãy (t_1, t_2, \dots, t_n) thành k đoạn để tổng đoạn lớn nhất là nhỏ nhất.

Ví dụ, với dãy $(10, 20, 30, 40)$ và $k = 2$ thì chia như sau $([10, 20, 30], [40])$ để tổng đoạn lớn nhất là nhỏ nhất bằng 60.

Input

- Dòng đầu chứa hai số nguyên dương n, k ($k \leq n$);
- Dòng thứ hai gồm n số nguyên dương t_1, t_2, \dots, t_n ($t_i \leq 10^9$).

Output

- Gồm một dòng chứa một số là tổng đoạn lớn nhất là nhỏ nhất.

Input	Output
4 2 10 20 30 40	60

Subtask 1: $n \leq 300$;

Subtask 2: $n \leq ?$;

book2

Alice có một quyển sách gồm n chương, chương thứ i ($1 \leq i \leq n$) có p_i trang. Alice dự định đọc quyển sách làm k đợt, mỗi đợt sẽ đọc một số chương liên tiếp nhau, giả sử trong một đợt Alice đọc s trang thì thời gian đọc cho đợt đó là s^2 đơn vị thời gian. Alice muốn tìm cách đọc để thời gian tổng thời gian đọc là nhỏ nhất.

Ví dụ, với dãy $(1, 2, 3, 4)$ và $k = 2$ thì chia như sau $([1, 2, 3], [4])$ để tổng thời gian đọc là nhỏ nhất là $6^2 + 4^2 = 52$.

Input

- Dòng đầu chứa hai số nguyên dương n, k ($k \leq n$);
- Dòng thứ hai gồm n số nguyên dương t_1, t_2, \dots, t_n ($t_i \leq 10^3$).

Output

- Gồm một dòng chứa một số là tổng thời gian đọc là nhỏ nhất.

Input	Output
4 2 1 2 3 4	52

Subtask 1: $n \leq 300$;

Subtask 2: $n \leq ?$;

Đoán số

An và Bình chơi trò đoán số. An nghĩ trong đầu một số trong phạm vi từ 1 đến n . Bình phải đoán số đó bằng cách đưa ra các câu hỏi dạng: “Số đó có nằm trong tập S không?”, trong đó S là tập các số bất kỳ trong phạm vi từ 1 tới n . Mỗi lần có câu trả lời khẳng định, Bình phải đưa cho An hai cái kẹo, trong trường hợp ngược lại phải đưa một cái.

Yêu cầu: Hãy xác định số kẹo tối thiểu Bình cần chuẩn bị để chắc chắn đoán được số An nghĩ.

Input

- Gồm một số nguyên n .

Output

- Số kẹo cần chuẩn bị.

Dữ liệu vào	Kết quả ra
2	2
6	5

Subtask 1: $n \leq 20$;

Subtask 2: $n \leq 200$;

Subtask 3: $n \leq 2 \times 10^5$;

Subtask 4: $n \leq 2 \times 10^9$;

mosaic4

Salma dự định tô màu một bức tranh khăm đất sét trên tường. Bức tranh là một lưới $N \times N$, được tạo thành từ N ô vuông 1×1 chưa tô màu. Các hàng của bức tranh được đánh số từ 0 đến $N - 1$ từ trên xuống dưới, và các cột được đánh số từ 0 đến $N - 1$ từ trái sang phải. Ô vuông ở hàng i và cột j ($0 \leq i < N, 0 \leq j < N$) được kí hiệu là (i, j) . Mỗi ô phải được tô màu trắng (kí hiệu là 0) hoặc đen (kí hiệu là 1).

Để tô màu cho bức tranh, trước tiên Salma chọn hai mảng X và Y có độ dài N , chỉ gồm các giá trị 0 và 1, thỏa mãn $X[0] = Y[0]$. Cô ấy tô màu các ô của hàng trên cùng (hàng 0) theo mảng X , sao cho màu của ô $(0, j)$ là $X[j]$ ($0 \leq j < N$). Cô ấy cũng tô màu các ô của cột ngoài cùng bên trái (cột 0) theo mảng Y , sao cho màu của ô $(i, 0)$ là $Y[i]$ ($0 \leq i < N$). Sau đó, cô ấy lặp lại các bước sau cho đến khi tất cả các ô đều được tô màu:

- ✓ Cô ấy tìm một ô (i, j) bất kì chưa tô màu mà có ô kè cạnh bên trên ($\hat{o}(i-1, j)$) và ô kè cạnh bên trái ($\hat{o}(i, j-1)$) đều đã được tô màu.
- ✓ Sau đó, cô ấy tô màu ô (i, j) thành màu đen nếu cả hai ô kè này đều có màu trắng; trái lại, cô ấy tô màu ô (i, j) thành màu trắng.

Có thể thấy rằng màu cuối cùng của các ô không phụ thuộc vào thứ tự các ô được Salma tô màu. Yasmin rất tò mò về màu của các ô trong bức tranh. Cô ấy hỏi Salma Q câu hỏi, được đánh số từ 0 đến $Q - 1$. Trong câu hỏi k ($0 \leq k < Q$), Yasmin chỉ định một hình chữ nhật con của bức tranh bởi:

- ✓ Hàng trên cùng $T[k]$ và hàng dưới cùng $B[k]$ ($0 \leq T[k] \leq B[k] < N$),
- ✓ Cột ngoài cùng bên trái $L[k]$ và cột ngoài cùng bên phải $R[k]$ ($0 \leq L[k] \leq R[k] < N$).

Câu trả lời cho câu hỏi là số ô màu đen trong hình chữ nhật con này. Cụ thể, Salma cần đếm có bao nhiêu ô (i, j) mà $T[k] \leq i \leq B[k], L[k] \leq j \leq R[k]$, và màu của ô (i, j) là màu đen. Hãy viết một chương trình trả lời các câu hỏi của Yasmin.

Các ràng buộc: $1 \leq N \leq 200\,000$; $1 \leq Q \leq 200\,000$; $X[i] \in \{0, 1\}$ và $Y[i] \in \{0, 1\}$ với mỗi i thỏa mãn $0 \leq i < N$; $X[0] = Y[0]$; $0 \leq T[k] \leq B[k] < N$ và $0 \leq L[k] \leq R[k] < N$ với mỗi k thỏa mãn $0 \leq k < Q$.

Input	Output
njJ9Z7VxxKGR6SUcJMgdzy3qMz4JZ1Tq	Wm5rkGNobnYjFI7TIY17RAm6FAQ2LlO9
N	OK
X[0] X[1] ... X[N-1]	C[0]
Y[0] Y[1] ... Y[N-1]	C[1]
Q	...
T[0] B[0] L[0] R[0]	C[S-1]
T[1] B[1] L[1] R[1]	
...	
T[Q-1] B[Q-1] L[Q-1] R[Q-1]	

BR

Anh em Bi và Ri cùng nhau chơi trò chơi như sau: Xếp ngẫu nhiên các viên bi màu xanh (B) là các viên bi của Bi và các viên bi màu đỏ (R) là các viên bi của Ri vào các hàng. Sau đó, hai anh em thay nhau thực hiện các lượt đi, Bi sẽ là người đi trước. Mỗi lượt đi, người chơi sẽ chọn viên bi của mình rồi loại bỏ viên bi đó cùng tất cả các viên bi nằm bên phải cùng hàng, ai không thực hiện được lượt đi sẽ là người thua cuộc.

Yêu cầu: Cho trạng thái các viên bi, hãy cho biết ai sẽ là người chiến thắng nếu cả hai đều biết cách chơi tối ưu.

Input

- Dòng đầu chứa số nguyên n là số hàng bi;
- Tiếp theo là n dòng, mỗi dòng chứa một xâu chỉ gồm 2 loại kí tự B hoặc R.

Output

- Ghi B nếu Bi thắng, ngược lại ghi R.

Dữ liệu vào	Kết quả ra
2 BRB RB	B

Subtask 1: $n \leq 5$; độ dài mỗi xâu không vượt quá 10;

Subtask 2: $n \leq 100$; độ dài mỗi xâu không vượt quá 100;

Phân số

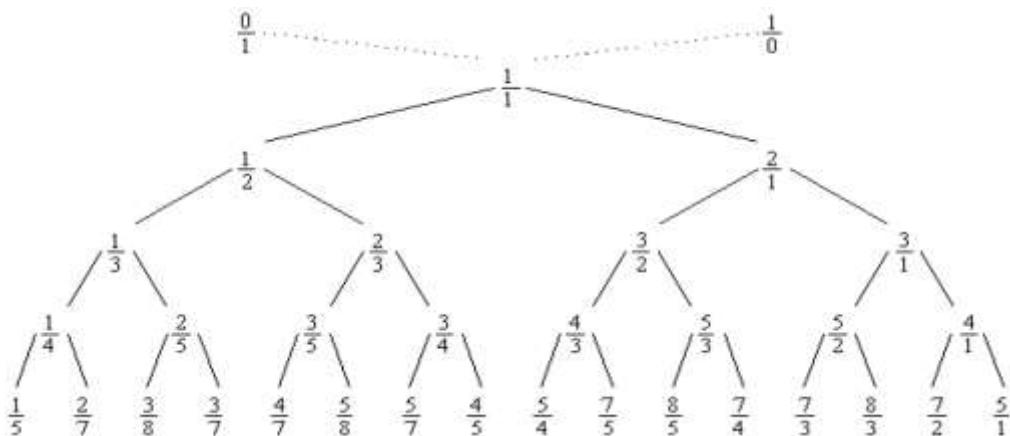
Với hai phân số $\frac{x}{y}$ và $\frac{u}{v}$ người ta tạo ra phân số $\frac{x+u}{y+v}$. Bắt đầu từ hai phân số $\frac{0}{1}$ và $\frac{1}{0}$, tiến hành tao ra các phân số.

Lượt đầu tiên tạo được các phân số $\frac{0}{1}; \frac{1}{1}; \frac{1}{0}$;

Lượt thứ hai tạo được các phân số $\frac{0}{1}; \frac{1}{2}; \frac{1}{1}; \frac{2}{1}; \frac{1}{0}$;

...

Thực hiện h lần để tạo được một cây phân số có gốc là phân số $\frac{1}{1}$ với độ cao $h - 1$. Ví dụ, trong hình dưới đây, với $h = 5$, không tính hai phân số $\frac{0}{1}$ và $\frac{1}{0}$ thì các phân số còn lại tạo thành một cây với độ cao 4.



Yêu cầu: Cho số nguyên dương h và k phân số $\frac{p_1}{q_1}, \frac{p_2}{q_2}, \dots, \frac{p_k}{q_k}$ trên cây tạo bởi h lượt, hãy tính số cạnh ít nhất trên cây cần dùng để liên thông được k phân số.

Dữ liệu:

- Dòng đầu chứa hai số nguyên h, k ;
- Dòng thứ i ($1 \leq i \leq k$) trong k dòng tiếp theo chứa hai số p_i, q_i ($p_i, q_i \leq 10^9$).

Kết quả: Gồm một dòng chứa một số nguyên là số cạnh ít nhất trên cây cần dùng để liên thông được k phân số.

Ví dụ:

Input	Output
5 4	
1 3	
2 5	
5 1	
1 5	

Ràng buộc:

- Có 25% số test ứng với 25% số điểm có $h \leq 5$ và $\frac{p_1}{q_1} = \frac{1}{1}$;
- Có 25% số test khác ứng với 25% số điểm có $h \leq 60; k \leq 2000$ và $\frac{p_1}{q_1} = \frac{1}{1}$;
- Có 20% số test khác ứng với 20% số điểm có $h \leq 60$ và $k \leq 5 \times 10^4$;
- Có 30% số test còn lại ứng với 30% số điểm có $h \leq 60$ và $k \leq 5 \times 10^5$.

Mã ẩn

Xét một dãy nhị phân A gồm n phần tử, phần tử A_i ($1 \leq i \leq n$) chỉ nhận giá trị 0 hoặc 1. Vì lí do bảo mật, ban đầu tất cả các phần tử của dãy A đều được ẩn giá trị và chỉ biết trong dãy A có chính xác k giá trị 0 ($0 \leq k \leq n$). Thời điểm 0 là thời điểm bắt đầu xác định dãy A . Có m thông tin về mối quan hệ giữa các cặp phần tử trong dãy A sẽ lần lượt xuất hiện, thông tin thứ t ($1 \leq t \leq m$) xuất hiện tại thời điểm t . Cấu trúc của thông tin thứ t được mô tả thông qua ba giá trị u_t, c_t, v_t . Trong đó, u_t, v_t tương ứng với chỉ số của hai phần tử trong dãy A ($1 \leq u_t < v_t \leq n$) và c_t là một trong các kí tự $>$, $<$ hoặc $=$ để biểu diễn mối quan hệ giữa A_{u_t} và A_{v_t} . Nếu c_t là kí tự $>$ thì $A_{u_t} > A_{v_t}$, nếu c_t là kí tự $<$ thì $A_{u_t} < A_{v_t}$, còn c_t là kí tự $=$ thì $A_{u_t} = A_{v_t}$. Các thông tin đều bảo đảm tính chính xác, hợp lí trên dãy A .

Cần tìm thời điểm s ($0 \leq s \leq m$) nhỏ nhất để xác định duy nhất một cách gán giá trị cho tất cả n phần tử của dãy A sao cho có chính xác k giá trị 0 và thỏa mãn s thông tin đầu tiên.

Yêu cầu: Cho các thông tin về dãy A , hãy đưa ra thời điểm s nhỏ nhất sao cho dãy A được xác định duy nhất. Trong trường hợp sử dụng tất cả m thông tin mà vẫn có nhiều hơn một cách gán giá trị cho tất cả n phần tử của dãy A thì đưa ra -1 .

Input

- Dòng đầu tiên chứa một số nguyên dương q là số lượng bộ dữ liệu;
- Tiếp theo gồm q nhóm dòng, mỗi nhóm mô tả một bộ dữ liệu theo khuôn dạng sau:
 - Dòng thứ nhất chứa ba số nguyên n, m và k cách nhau bởi dấu cách ($1 \leq m \leq 4 \times 10^5$; $0 \leq k \leq n$);
 - Dòng thứ t trong số m dòng tiếp theo ($1 \leq t \leq m$) chứa ba giá trị u_t, c_t, v_t mô tả thông tin xuất hiện tại thời điểm t . Các giá trị cách nhau đúng một dấu cách.

Dữ liệu vào đảm bảo tồn tại ít nhất một dãy A có chính xác k giá trị 0 thỏa mãn tất cả m thông tin. Tổng các số m trong q bộ dữ liệu không quá 2×10^6 .

Output

- Mỗi dòng chứa một số nguyên s tương ứng với thời điểm nhỏ nhất để xác định duy nhất một cách gán giá trị cho tất cả n phần tử của dãy A trong dữ liệu vào tương ứng. Nếu không tìm được thời điểm s thỏa mãn thì ghi ra -1 .

Gọi N là tổng các số n trong q bộ dữ liệu.

Subtask 1: $1 \leq n \leq 2 \times 10^3$ và $N \leq 10^4$;

Subtask 2: $1 \leq n \leq 2 \times 10^4$ và $N \leq 10^5$;

Dữ liệu	Kết quả	Giải thích
3	5	Trong dãy thứ nhất, tại thời điểm $s = 5$ xác định duy nhất dãy $\{1, 0, 1, 0, 1, 0\}$ thỏa mãn các thông tin tính đến thời điểm 5. Với thời điểm $t = 4$ có thêm ít nhất một dãy $\{1, 0, 0, 1, 0, 1\}$ thỏa mãn các thông tin tính đến thời điểm 4.
6 6 3	0	Trong dãy thứ hai, không cần xét đến các thông tin bổ sung mà vẫn xác định dãy duy nhất $\{0, 0, 0, 0\}$. Do đó, đưa ra s bằng 0.
1 > 2	-1	Trong dãy thứ ba, mặc dù sử dụng hết 2 thông tin nhưng vẫn có ít nhất hai dãy thỏa mãn tất cả 2 thông tin là: $\{0, 0, 1, 1\}$ và $\{1, 1, 0, 0\}$. Do đó, đưa ra -1 .
3 = 5		
4 = 6		
1 > 2		
2 = 4		
1 = 3		
4 2 4		
1 = 2		
3 = 4		
4 2 2		
1 = 2		
3 = 4		

Xếp hàng

Để trình diễn một tiết mục trong màn khai mạc Đại hội thể thao quốc tế, đạo diễn Hùng đã mời n vận động viên có chiều cao khác nhau từng đôi tham gia. Theo kịch bản, n vận động viên sẽ được xếp thành một hàng dọc, đầu hàng ở phía khán đài A (khán đài chứa các khách mời quốc tế), cuối hàng ở phía khán đài B (khán đài có nhiều du khách và các quan chức địa phương). Đạo diễn muốn rằng từ phía khán đài A, khán giả có thể nhìn thấy P vận động viên, còn từ phía khán đài B khán giả có thể nhìn thấy Q vận động viên. Một vận động viên được nhìn thấy từ phía khán đài A nếu như tất cả các vận động viên đứng trước (theo chiều từ đầu hàng đến cuối hàng) đều có chiều cao thấp hơn vận động viên này. Một vận động viên được nhìn thấy từ phía khán đài B nếu như tất cả các vận động viên đứng sau (theo chiều từ đầu hàng đến cuối hàng) đều có chiều cao thấp hơn vận động viên này.

Ví dụ: Có 9 vận động viên được xếp theo thứ tự với chiều cao tương ứng là

3, 2, 4, 1, 9, 8, 7, 5, 6

thì từ phía đầu hàng (ở phía bên trái) có thể nhìn thấy 3 người (với chiều cao là 3, 4, 9), còn từ phía cuối hàng (ở phía bên phải) có thể nhìn thấy 4 người (với chiều cao là 6, 7, 8, 9).

Yêu cầu: Hãy giúp đạo diễn xác định xem có bao nhiêu cách xếp n vận động viên thành hàng dọc thoả mãn điều kiện đặt ra.

Input

- Dòng đầu tiên ghi ba số nguyên dương n, P, Q ($P, Q \leq n$);
- Dòng thứ hai gồm n số nguyên dương là các độ cao của n vận động viên được mời tham gia thực hiện tiết mục trình diễn.

Hai số liên tiếp trên cùng một dòng được ghi cách nhau bởi ít nhất một dấu cách.

Output

- Gồm một số nguyên là phần dư trong phép chia số lượng cách xếp tìm được cho $10^9 + 7$.

Input	Output
3 2 1	
1 2 3	1

Giải thích: Trong số 6 cách xếp 3 vận động viên thành một hàng dọc, có một hàng duy nhất các vận động viên được xếp theo thứ tự chiều cao là 2, 1, 3 thoả mãn yêu cầu đặt ra.

Subtask 1: $n \leq 10$;

Subtask 2: $n \leq 500$; $Q = 1$;

Subtask 3: $n \leq 500$;

Subtask 4: $n \leq 2000$;

Chia kẹo

Alice có n chiếc kẹo và muốn chia n chiếc kẹo vào các túi để trao thưởng cho các bạn nhỏ, mỗi túi sẽ tương ứng với một mức thưởng. Cụ thể, với hai số nguyên dương n, k , Alice muốn chia kẹo thỏa mãn:

- ✓ Số lượng túi ít nhất là 2 túi;
- ✓ Tổng số kẹo trong các túi bằng đúng n ;
- ✓ Chênh lệch số kẹo giữa hai túi bất kì lớn hơn k .

Ví dụ, với $n = 9, k = 1$, Alice có 4 cách chia như sau: $1 + 8; 2 + 7; 3 + 6; 1 + 3 + 5$.

Yêu cầu: Cho ba số nguyên dương n, k, M . Gọi C là số cách chia kẹo thỏa mãn, hãy tính $C \% M$, trong đó $\%$ là phép toán chia lấy dư.

Dữ liệu: Gồm một dòng chứa ba số nguyên dương n, k, M ($k \leq n \leq 10^5; M \leq 10^9 + 9$).

Kết quả: Gồm một số nguyên là phần dư trong phép chia C cho M .

Ràng buộc:

- Có 30% số test ứng với 30% số điểm của bài có $n \leq 20$;
- Có 30% số test khác ứng với 30% số điểm của bài có $n \leq 200$;
- Có 20% số test khác ứng với 20% số điểm của bài có $n \leq 2000$;
- Có 20% số test còn lại ứng với 20% số điểm của bài có $n \leq 10^5$.

Ví dụ:

Input	Output
9 1 100	4

Liên thông

Một khu du lịch do Alice quản lý là một quần đảo được mô tả bằng một bảng kí tự gồm m hàng và n cột, các hàng được đánh số từ 1 đến m từ trên xuống dưới, các cột được đánh số từ 1 đến n từ trái sang phải. Ô nằm giao giữa hàng i và cột j gọi là ô (i, j) và chứa kí tự '1' hoặc '0', trong đó '1' mô tả vùng đất (gọi là ô đất), '0' mô tả vùng nước (gọi là ô nước). Một đảo là một vùng liên thông gồm các ô đất mà hai ô có thể đi đến nhau thông qua các ô đất chung cạnh. Khách du lịch có thể chọn xuất phát tại một ô đất và đi bộ thăm hết các ô đất trên đảo (chứa ô đất xuất phát) bằng cách di chuyển thông qua các ô chung cạnh.

Với mong muốn khách du lịch có thể đi bộ và thăm quan được nhiều hơn, Alice lên kế hoạch dựng cầu treo trên các ô nước để liên thông một số đảo. Cụ thể, sau khi dựng cầu treo ở các ô nước, du khách có thể di chuyển vào các ô nước đó như đi vào các ô đất, Alice mong muốn tồn tại cách thăm quan cho du khách mà tổng số ô đất du khách có thể đi thăm quan được lớn hơn hoặc bằng s và số lượng ô nước cần dựng cầu treo là ít nhất.

Yêu cầu: Cho bảng kí tự mô tả quần đảo và T phương án tương ứng với T số nguyên dương s_1, s_2, \dots, s_T , với mỗi phương án s_k ($1 \leq k \leq T$) hãy giúp Alice tính số ô nước ít nhất cần dựng cầu treo để tổng số ô đất mà khách du lịch có thể đi bộ và thăm được lớn hơn hoặc bằng s_k .

Dữ liệu:

- Dòng đầu gồm ba số nguyên dương m, n, T ($T \leq 100$);
- Dòng thứ i ($1 \leq i \leq m$) trong số m dòng tiếp theo chứa một xâu độ dài n chỉ gồm hai loại kí tự '1' và '0';
- Dòng cuối cùng chứa T số nguyên dương s_1, s_2, \dots, s_T ($s_k \leq m \times n$).

Kết quả:

- Gồm T số nguyên tương ứng với T phương án, mỗi số là số ô nước ít nhất cần dựng cầu treo hoặc đưa ra -1 nếu không tồn tại cách dựng thỏa mãn.

Ràng buộc:

Gọi c là số đảo trong khu du lịch, trong tất cả các test đều thỏa mãn $1 \leq c \leq 10$ và $m, n \leq 100$, ngoài ra:

- Có 10% số test ứng với 10% số điểm của bài có $c = 1; m, n \leq 30$;
- Có 25% số test ứng khác với 25% số điểm của bài có $c \leq 2; m, n \leq 30$;
- Có 25% số test ứng khác với 25% số điểm của bài có $c \leq 3; m, n \leq 30$;
- Có 20% số test ứng khác với 20% số điểm của bài $c \leq 8; m, n \leq 30$;
- Có 20% số test còn lại ứng với 20% số điểm của bài không có ràng buộc nào thêm.

Ví dụ:

Input	Output	Input	Output
5 6 3 000001 011101 010011 000001 001100 9 10 15	1 2 -1	3 5 6 10101 01010 10101 1 2 3 4 5 6	0 1 1 1 2 2

Chiếu sáng

Quảng trường trong khu du lịch mà Alice quản lí là một sân rộng hình vuông với tọa độ góc trái dưới là $(0,0)$ và góc phải trên là $(10^6, 10^6)$. Có n máy chiếu sáng laser, máy thứ i ($1 \leq i \leq n$) khi chiếu sẽ xuất hiện một vệt sáng là một đoạn thẳng có tọa độ đầu mút là (x_i, y_i) và (u_i, v_i) trên quảng trường. Biết rằng, không có hai máy nào khi cùng chiếu sáng tạo ra hai vệt sáng là hai đoạn thẳng thỏa mãn: hai đoạn thẳng cùng nằm trên một đường thẳng và có điểm chung. Alice muốn tạo ra các kịch bản chiếu sáng, cô cần đếm số cách chọn 4 máy để khi cả 4 máy cùng chiếu sáng thì sẽ xuất hiện một hình chữ nhật tạo bởi các vệt sáng với 4 góc là 4 giao điểm của 4 đoạn thẳng.

Yêu cầu: Cho thông tin về n đoạn thẳng mô tả vệt sáng của các máy chiếu laser, hãy giúp Alice đếm số cách chọn thỏa mãn.

Dữ liệu:

- Dòng đầu chứa số nguyên dương n ($n \leq 5000$);
- Dòng thứ i ($1 \leq i \leq n$) trong n dòng sau chứa bốn số x_i, y_i, u_i, v_i ($0 \leq x_i, y_i, u_i, v_i \leq 10^6$).

Kết quả: Gồm một dòng chứa một số nguyên không âm là số cách chọn thỏa mãn.

Ràng buộc:

- Có 20% số test ứng với 20% số điểm của bài thỏa mãn: $n \leq 50$ và các đoạn thẳng đều song song hoặc vuông góc với cạnh quảng trường;
- Có 15% số test khác ứng với 15% số điểm của bài thỏa mãn: $n \leq 50$;
- Có 20% số test khác ứng với 20% số điểm của bài thỏa mãn: $n \leq 500$ và các đoạn thẳng đều song song hoặc vuông góc với cạnh quảng trường;
- Có 10% số test khác ứng với 10% số điểm của bài thỏa mãn: $n \leq 500$;
- Có 20% số test khác ứng với 20% số điểm của bài thỏa mãn: Các đoạn thẳng đều song song hoặc vuông góc với cạnh quảng trường;
- Có 15% số test còn lại ứng với 15% số điểm của bài mãn: Không có ràng buộc nào thêm.

Ví dụ:

Input	Output	Minh họa
10 30 20 30 80 10 20 100 20 10 40 100 40 10 60 30 60 10 20 10 80 50 60 100 60 30 40 50 60 30 40 50 20 90 20 50 60 90 60 50 20	4	

Truy vấn đoạn

Có n đoạn số nguyên trong đoạn $[1, M]$ nhưng bạn không biết chính xác các đoạn. Bạn cần tìm số đoạn độc lập rời nhau nhiều nhất bằng cách sử dụng không quá q truy vấn $\text{query}(L, R)$, hàm này trả về số đoạn nằm trong đoạn $[L, R]$.

Cụ thể, bạn cần xây dựng hàm `int solve(n, q)`, hàm trả về số đoạn độc lập rời nhau nhiều nhất, hàm này có thể gọi hàm dưới đây:

- `int query(L, R)`, hàm trả số đoạn $[a, b]$ trong n đoạn mà $L \leq a \leq b \leq R$.

Chú ý, hàm này không adaptive.

Subtask 1: $n \leq 1000$; $M \leq 10^3$; $q = 10^6$

Subtask 2: $n \leq 25000$; $M \leq 10^5$; $q = 10^5$

Subtask 3: $n \leq 25000$; $M \leq 10^9$; $q = 7.5 \times 10^5$

Subtask 4: $n \leq 25000$; $M \leq 10^9$; $q = 4.4 \times 10^5$

File lrq.h

```
// Cần cài đặt hàm
int solve(int n, int q);

// Sử dụng hàm
int query(int L, int R);
```

Trang trại

Alice đang lên kế hoạch dựng các chuồng cho n chú bò trong trang trại của cô. Các chú bò được đánh số từ 1 đến n , chú bò thứ i ($1 \leq i \leq n$) có chiều cao là h_i . Mỗi chuồng chứa tối đa hai chú bò và nếu muốn dựng chuồng chứa được chú bò có chiều cao s thì mất chi phí là s . Cụ thể: Nếu muốn dựng một chuồng cho một chú bò có chiều cao s thì mất chi phí s ; Nếu muốn dựng một chuồng cho hai chú bò có chiều cao là s_1, s_2 thì mất chi phí là $\max(s_1, s_2)$. Vì lí do bảo đảm sức khỏe cho chú bò có chiều cao thấp hơn nên hai chú bò cùng chuồng có chênh lệch chiều cao không được vượt quá d . Alice mong muốn xây dựng các chuồng cho n chú bò với tổng chi phí ít nhất.

Yêu cầu: Cho dãy số nguyên dương h_1, h_2, \dots, h_n và T trường hợp thử nghiệm d_1, d_2, \dots, d_T , với trường hợp thử nghiệm thứ j ($1 \leq j \leq T$), hãy tính tổng chi phí ít nhất để xây dựng các chuồng cho n chú bò mà hai chú bò cùng một chuồng có chênh lệch chiều cao không vượt quá d_j .

Dữ liệu:

- Dòng đầu chứa hai số nguyên dương n, T ;
- Dòng thứ hai chứa n số nguyên dương h_1, h_2, \dots, h_n ($h_i \leq 10^9$);
- Dòng thứ ba chứa T số nguyên dương d_1, d_2, \dots, d_T ($d_j \leq 10^9$).

Kết quả: Ghi ra T dòng, dòng thứ j ($1 \leq j \leq T$) chứa một số nguyên là tổng chi phí ít nhất để dựng các chuồng cho n chú bò của trường hợp thử nghiệm thứ j .

Ví dụ:

Input	Output	Giải thích
4 2	9	Với $d_1 = 1$, dựng 3 chuồng với tổng chi phí: $1 + \max(2, 3) + 5 = 9$.
1 3 5 2	7	Với $d_2 = 2$, dựng 2 chuồng với tổng chi phí: $\max(1, 2) + \max(3, 5) = 7$.
1 2		

Ràng buộc:

- Có 10% số test của bài có $n = 2; T = 1$;
- Có 30% số test khác của bài có $n \leq 10; T = 1$;
- Có 40% số test khác của bài có $n \leq 1000; T = 100$;
- Có 20% số test còn lại của bài có $n \leq 10^5; T \leq 10^5$.

Giao hữu bóng đá

Có n đội bóng đá (các đội được đánh số từ 1 đến n) thi đấu giao hữu theo thể thức vòng tròn, mỗi đội sẽ thi đấu với tất cả các đội còn lại. Để tăng sự hấp dẫn, Ban tổ chức quyết định mỗi trận đấu sẽ luôn được phân định thắng thua, cụ thể nếu trong thời gian thi đấu mà hai đội không phân thắng thua thì sẽ được phân định bằng đá luân lưu. Sau khi kết thúc giải đấu, với cặp số giới hạn L, R ($1 \leq L < R \leq n$), Ban tổ chức muốn thống kê một trong hai thông tin thú vị như sau:

- Chuỗi thắng dài nhất: Cần chọn một danh sách nhiều nhất các đội phân biệt p_1, p_2, \dots, p_r mà $L \leq p_1, p_2, \dots, p_r \leq R$ và đội trước thắng đội liền sau trong danh sách.
- Vòng thắng nhỏ nhất: Cần chọn một danh sách ít nhất các đội phân biệt q_1, q_2, \dots, q_s mà $L \leq q_1, q_2, \dots, q_r \leq R$ và đội trước thắng đội liền sau trong danh sách, ngoài ra đội cuối cùng trong danh sách lại thắng đội đầu tiên trong danh sách.

Yêu cầu: Cho biết kết quả tất cả các trận đấu của n đội biểu diễn bằng mảng c , trong đó $c_{ij} = 0$ nghĩa là đội i thua đội j ($i \neq j$), trái lại $c_{ij} = 1$ cho biết đội i thắng đội j ($c_{ji} = 1 - c_{ij}$; $c_{ii} = 0$), T cặp số giới hạn $L_1, R_1, L_2, R_2, \dots, L_T, R_T$ và số nguyên k , với mỗi cặp số giới hạn nếu $k = 1$ đưa ra thông tin về chuỗi thắng dài nhất, nếu $k = 2$ đưa ra thông tin về vòng thắng nhỏ nhất thỏa mãn.

Dữ liệu:

- Dòng đầu tiên ghi hai số nguyên dương n, T, k ($n \leq 1000; T \leq 100; 1 \leq k \leq 2$);
- Dòng thứ hai chứa T cặp số $L_1, R_1, L_2, R_2, \dots, L_T, R_T$ mô tả T cặp số giới hạn;
- Tiếp theo là n dòng, trong đó số thứ j ($1 \leq j \leq n$) của dòng thứ i ($1 \leq i \leq n$) chứa số nguyên c_{ij} .

Kết quả: Ghi ra T dòng, mỗi dòng tương ứng câu trả lời cho một cặp giới hạn có khuôn dạng: Số đầu tiên ghi -1 nếu không tìm được dãy các đội thỏa mãn, ngược lại ghi số lượng đội trong danh sách tìm được; Nếu có dãy thỏa mãn, tiếp theo ghi chỉ số các đội lần lượt trong danh sách.

Ví dụ:

Input	Output	Hình minh họa
4 1 1 1 4 0 0 0 1 1 0 0 0 1 1 0 0 0 1 1 0	4 2 1 4 3	<pre> graph TD 1((1)) --> 2((2)) 1((1)) --> 3((3)) 2((2)) --> 4((4)) 3((3)) --> 4((4)) 4((4)) --> 1((1)) </pre>
4 2 2 1 4 1 3 0 0 0 1 1 0 0 0 1 1 0 0 0 1 1 0	3 1 4 2 -1	<pre> graph TD 1((1)) --> 2((2)) 1((1)) --> 3((3)) 2((2)) --> 2((2)) 2((2)) --> 4((4)) 3((3)) --> 4((4)) 4((4)) --> 1((1)) </pre>

Ràng buộc:

- Có 10% số test của bài có $n = 3; T \leq 3$;
- Có 20% số test khác của bài có $n \leq 20; T \leq 3$ và $k = 1$;
- Có 20% số test khác của bài có $n \leq 20; T \leq 3$ và $k = 2$;
- Có 25% số test khác của bài có $k = 1$;
- Có 25% số test còn lại của bài có $k = 2$.

Dàn đèn trang trí

Một dàn đèn trang trí gồm n đèn được đánh số từ 1 đến n và $n - 1$ đoạn dây nối điều khiển, mỗi đoạn nối một cặp hai đèn khác nhau. Hệ thống dây nối điều khiển thỏa mãn tính chất sau đây: Không có hai đoạn dây nào nối cùng một cặp đèn và hơn nữa không tìm được dây các đèn $v_1, v_2, \dots, v_k, v_1$ ($k > 2$), trong đó hai đèn liên tiếp là có đoạn dây nối. Tại mỗi thời điểm, từng đèn sẽ sáng màu xanh hoặc đỏ. Bộ điều khiển hệ thống đèn có thể thực hiện tác động nhiều lần việc thay đổi trạng thái các đèn, mỗi lần tác động là thay đổi màu của một đèn nào đó và tất cả các đèn có dây nối với nó, cụ thể nếu đèn đang sáng màu xanh sẽ chuyển sang sáng màu đỏ, ngược lại nếu đèn đang sáng màu đỏ sẽ chuyển sang sáng màu xanh. Vì lý do kỹ thuật, hệ thống đã bỏ sung thêm một đoạn dây nối giữa hai đèn khau nhau x, y (giữa hai đèn này chưa có dây nối trước đó).

Yêu cầu: Cho biết trạng thái ban đầu màu của n đèn và thông tin về các dây nối điều khiển ban đầu và dây nối bổ sung, hãy tìm cách điều khiển để tắt cả các đèn sáng màu xanh.

Dữ liệu:

- Dòng đầu chứa hai số nguyên dương n, T là số lượng đèn và số trường hợp thử nghiệm;
- Dòng thứ k trong $n - 1$ dòng tiếp theo chứa thông tin về đoạn dây nối điều khiển thứ k bao gồm hai số nguyên u_k, v_k , trong đó u_k, v_k là chỉ số của hai đèn là các đầu mút của đoạn dây nối điều khiển thứ k ($1 \leq k \leq n - 1$);
- Tiếp theo là một dòng chứa hai số nguyên x, y ;
- Dòng thứ i ($1 \leq i \leq T$) trong T dòng cuối chứa n số $c_{i1}, c_{i2}, \dots, c_{in}$, trong đó c_{ij} là màu của đèn thứ j ($1 \leq j \leq n$) trong trường hợp thử nghiệm thứ i ($c_{ij} = 1$ nếu đèn j sáng màu xanh và $c_{ij} = 0$ nếu đèn j sáng màu đỏ).

Kết quả: Ghi ra gồm T dòng, mỗi dòng là phương án điều khiển cho trường hợp thử nghiệm tương ứng trong dữ liệu vào, theo khuôn dạng sau: Ghi -1 nếu không tồn tại cách điều khiển thỏa mãn. Ngược lại số đầu tiên của dòng là số s ; tiếp theo là s số l_1, l_2, \dots, l_s mô tả cách điều khiển, trong đó tác động thứ h ($1 \leq h \leq s$) làm đảo màu của đèn l_h và các đèn có dây nối với l_h .

Ví dụ:

Input	Output	Minh họa
4 1	1 4	
1 2		
2 3		
3 4		
1 4		
0 1 0 0		

Ràng buộc:

- Có 30% số test của bài có $n \leq 20; T \leq 5$;
- Có 40% số test khác của bài có $n \leq 300; T \leq 50$;
- Có 30% số test còn lại của bài có $n \leq 3000; T \leq 500$.

Nilei4

Bạn muôn vận chuyển N cỗ vật qua sông Nile. Các cỗ vật được đánh số từ 0 đến N – 1. Khối lượng của cỗ vật i ($0 \leq i < N$) là W[i].

Để vận chuyển các cỗ vật, bạn dùng nhiều thuyền chuyên dụng. Mỗi chiếc thuyền có thể mang nhiều nhất hai cỗ vật.

- ✓ Nếu bạn quyết định đặt duy nhất một cỗ vật lên một chiếc thuyền, khối lượng của cỗ vật có thể nặng tùy ý.
- ✓ Nếu bạn muốn đặt hai cỗ vật trên cùng một chiếc thuyền, bạn phải chắc chắn rằng chiếc thuyền được cân bằng. Cụ thể, bạn có thể vận chuyển cỗ vật p và q ($0 \leq p < q < N$) trên cùng một chiếc thuyền chỉ khi chênh lệch tuyệt đối giữa khối lượng của chúng nhiều nhất là D, nghĩa là $|W[p] - W[q]| \leq D$.

Để vận chuyển một cỗ vật, bạn phải trả chi phí phụ thuộc vào số lượng cỗ vật trên cùng một chiếc thuyền đó. Chi phí để vận chuyển cỗ vật i ($0 \leq i < N$) là: A[i] nếu bạn đặt duy nhất cỗ vật này trên chiếc thuyền mang nó, hoặc B[i] nếu bạn đặt cỗ vật này trên thuyền cùng với một cỗ vật khác. Lưu ý rằng, trong trường hợp thứ hai, bạn phải trả chi phí cho cả hai cỗ vật trên chiếc thuyền đó. Cụ thể, nếu bạn quyết định vận chuyển cỗ vật p và q ($0 \leq p < q < N$) trên cùng một chiếc thuyền, bạn phải trả chi phí $B[p] + B[q]$. Vận chuyển duy nhất một cỗ vật trên một chiếc thuyền luôn luôn có chi phí đắt hơn vận chuyển cỗ vật đó cùng với một cỗ vật khác trên thuyền, tức là $B[i] < A[i]$ với mọi i thoả mãn $0 \leq i < N$.

Không may, dòng sông rất khó dự đoán và giá trị D thay đổi thường xuyên. Nhiệm vụ của bạn là trả lời Q câu hỏi, đánh số từ 0 đến Q – 1. Các câu hỏi được mô tả bằng một mảng E có độ dài Q. Câu trả lời cho câu hỏi j ($0 \leq j < Q$) là tổng chi phí nhỏ nhất để vận chuyển tất cả N cỗ vật khi giá trị D bằng E[j].

Ràng buộc:

$$1 \leq N \leq 100\,000; 1 \leq Q \leq 100\,000; 1 \leq W[i] \leq 10^9; 1 \leq B[i] < A[i] \leq 10^9; 1 \leq E[j] \leq 10^9;$$

Input	Output
1U22aPSRkiGgB7zXbfJs9IqSfacMQ9t7	4FBwMMwcdlgPodEtp0owUlxQgA5SuIlq
N	OK
W[0] A[0] B[0]	R[0]
W[1] A[1] B[1]	R[1]
...	...
W[N-1] A[N-1] B[N-1]	R[S-1]
Q	
E[0]	
E[1]	
...	
E[Q-1]	