

06/11:

## Bài A. MTC

File dữ liệu vào:	<b>stdin</b>
File kết quả:	<b>stdout</b>
Hạn chế thời gian:	1 giây
Hạn chế bộ nhớ:	512MB

Dãy  $X = x_1, x_2, \dots, x_k$  được gọi là núi nếu tồn tại số tự nhiên  $i$  sao cho:

- $x_j < x_{j+1} \forall 1 \leq j < i$ ;
- $x_j > x_{j+1} \forall i \leq j < k$ .

Cho dãy số nguyên  $a$ , hãy đếm số cách xóa đi một số phần tử của  $a$  sao cho dãy còn lại là núi. Hai cách xóa được cho là khác nhau nếu tồn tại một vị trí được xóa trong cách này nhưng không được xóa trong cách kia.

### Dữ liệu vào

- Dòng đầu chứa số phần tử của dãy  $a$ :  $n$ ;
- Dòng tiếp theo chứa dãy  $a$ .

### Kết quả

In ra phần dư của số cách xóa khi chia cho  $10^9 + 7$ .

### Ví dụ

stdin	stdout
5 1 2 3 2 1	27

### Giải thích

Các cách xóa thỏa mãn là (dãy con sau khi xóa):  $\emptyset, 1, 2, 3, 2, 1, 12, 13, 12, 23, 21, 32, 31, 21, 123, 121, 132, 131, 121, 232, 231, 321, 1232, 1231, 1321, 2321, 12321$ .

### Hạn chế

- Trong tất cả các test:  $n \leq 10^5$ .  $-10^9 \leq a_i \leq 10^9$ ;
- Có 30% test với  $n \leq 20$ ;
- Có 30% test với  $20 < n \leq 1000$ ;
- Có 40% test với ràng buộc gốc.

## Bài B. MOSUM

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

Có một cuộc thi chạy đặc biệt đang diễn ra trên hành tinh xyz. Có  $n$  vận động viên, đánh số từ 1 đến  $n$ . Vận động viên thứ  $i$  mất  $t_i$ (s) để chạy một vòng quanh sân chạy. Cuộc thi này đặc biệt là vì:

- Mỗi lần chỉ có 2 người chạy. Để xem ai nhanh hơn, họ sẽ xuất phát cùng lúc tại vạch xuất phát (cũng là vạch đích) và chạy một hoặc nhiều vòng cho đến khi cả hai người chạm vạch đích vào cùng một thời điểm. Lúc này ai chạy được nhiều vòng hơn sẽ là người nhanh hơn
- Tất cả các cặp đều thi với nhau, tức là có  $\frac{n(n-1)}{2}$  lần chạy, sau đó tính điểm để phân thứ hạng

Giả sử thời gian chuyển giữa các cặp thi là không đáng kể, tính tổng thời gian diễn ra cuộc thi

### Dữ liệu vào

- Dòng đầu ghi số nguyên  $n$
- Dòng thứ hai ghi  $n$  số nguyên  $t_i$

### Kết quả

- Ghi kết quả sau khi chia lấy dư cho  $10^9 + 7$

### Ví dụ

stdin	stdout
3	
1 2 3	11

### Hạn chế

- Subtask 0:  $1 \leq n, t_i \leq 10^5$
- Có 30% test với  $n \leq 1000$

## Bài C. WATER

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

Chỗ ở của Đỗ Nam Trung (nhà xanh) có dạng một hình tròn. Có  $n$  cây xanh được trồng xung quanh đó, để thuận tiện các cây sẽ được đánh số từ 1 đến  $n$  ngược chiều kim đồng hồ. Tại tâm của hình tròn ông đặt một máy tưới nước, máy này được lập trình để tưới nước cho tất cả các cây. Có  $m$  lệnh sẽ được thực hiện tuần tự, lệnh thứ  $i$  là tưới các cây theo thứ tự ngược chiều kim đồng hồ, bắt đầu từ cây được đánh số  $l_i$  đến cây  $r_i$ .

Sau khi đến Việt Nam và tiếp xúc với các coder, Trung nhận ra dãy lệnh của ông có thể dư thừa. Ông muốn xóa bỏ một số nhiều nhất các lệnh sao cho dãy lệnh còn lại vẫn đảm bảo tưới hết cho các cây. Hãy giúp ông!

### Dữ liệu vào

- Dòng đầu chứa hai số nguyên dương:  $n \ m$
- $m$  dòng tiếp theo, dòng thứ  $i$  chứa:  $l_i \ r_i$

### Kết quả

- Ghi một số nguyên là số lệnh có thể xóa. In ra -1 nếu Trung không thể tưới hết các cây của mình

### Ví dụ

stdin	stdout
6 3	
5 6	
2 5	
4 1	1

### Hạn chế

- $1 \leq n, m \leq 10^5$
- Có 30% số test với  $m \leq 5000$

## Bài D. RADIO

File dữ liệu vào: **RADIO.inp**  
File kết quả: **RADIO.out**  
Hạn chế thời gian: 1 giây

Năm 2050, loài người có bước đột phá trong việc chinh phục vũ trụ. Họ xây dựng được  $n$  trạm không gian thẳng hàng, cách đều nhau 1s ánh sáng. Các trạm không gian được đánh số từ 1 đến  $n$ , xuất phát từ trạm 1 là trái đất, theo thứ tự xa dần trái đất (các trạm nằm về cùng 1 phía so với trái đất)

Do đặc điểm về cách mã hóa thông tin của các trạm, trạm không gian thứ  $i$  có khả năng nhận tín hiệu radio từ các trạm có số hiệu  $Ll_i$  đến  $Lr_i$  ( $1 \leq Ll_i \leq Lr_i \leq i$ ), và gửi tín hiệu radio đến các trạm có số hiệu từ  $Rl_i$  đến  $Rr_i$  ( $i \leq Rl_i \leq Rr_i \leq n$ ). Tốc độ gửi đi của sóng radio trong không gian là tốc độ ánh sáng. Thời gian cần thiết để trạm thứ  $i$  xử lý tín hiệu (giải mã tín hiệu nhận được thành thông tin hoặc mã hóa thông tin gửi đi thành tín hiệu radio) là  $t_i$ (s). Điều này có nghĩa, để có thể gửi thông tin từ trạm  $i$  sang trạm  $j$  ( $i < j$ ,  $Ll_j \leq i \leq Lr_j$  và  $Rl_i \leq j \leq Rr_i$ ) thì trạm  $i$  cần mã hóa thông tin, gửi tín hiệu sau khi mã hóa sang trạm  $j$ , và giải mã tín hiệu nhận được ở trạm  $j$ , do đó mất thời gian  $t_i + j - i + t_j$ (s)

**Yêu cầu:** Tính toán thời gian ít nhất để gửi thông tin từ trái đất đến các trạm khác

### Dữ liệu vào

Năm 2050, loài người có bước đột phá trong việc chinh phục vũ trụ. Họ xây dựng được  $n$  trạm không gian thẳng hàng, cách đều nhau 1s ánh sáng. Các trạm không gian được đánh số từ 1 đến  $n$ , xuất phát từ trạm 1 là trái đất, theo thứ tự xa dần trái đất (các trạm nằm về cùng 1 phía so với trái đất)

Do đặc điểm về cách mã hóa thông tin của các trạm, trạm không gian thứ  $i$  có khả năng nhận tín hiệu radio từ các trạm có số hiệu  $Ll_i$  đến  $Lr_i$  ( $1 \leq Ll_i \leq Lr_i \leq i$ ), và gửi tín hiệu radio đến các trạm có số hiệu từ  $Rl_i$  đến  $Rr_i$  ( $i \leq Rl_i \leq Rr_i \leq n$ ). Tốc độ gửi đi của sóng radio trong không gian là tốc độ ánh sáng. Thời gian cần thiết để trạm thứ  $i$  xử lý tín hiệu (giải mã tín hiệu nhận được thành thông tin hoặc mã hóa thông tin gửi đi thành tín hiệu radio) là  $t_i$ (s). Điều này có nghĩa, để có thể gửi thông tin từ trạm  $i$  sang trạm  $j$  ( $i < j$ ,  $Ll_j \leq i \leq Lr_j$  và  $Rl_i \leq j \leq Rr_i$ ) thì trạm  $i$  cần mã hóa thông tin, gửi tín hiệu sau khi mã hóa sang trạm  $j$ , và giải mã tín hiệu nhận được ở trạm  $j$ , do đó mất thời gian  $t_i + j - i + t_j$ (s)

**Yêu cầu:** Tính toán thời gian ít nhất để gửi thông tin từ trái đất đến các trạm khác

### Kết quả

Ghi  $n - 1$  số nguyên trên 1 dòng, là thời gian cần thiết để gửi thông tin từ trái đất đến trạm  $2, 3, \dots, n$ . Với những trạm không thể gửi thông tin đến thì ghi -1

### Ví dụ

RADIO.inp	RADIO.out
5	10 5 7 11
2 1 1 2 4	
7 1 1 5 5	
1 1 2 4 5	
0 2 3 4 5	
3 1 4 5 5	

### Hạn chế

- $1 \leq n \leq 10^5$ ,  $0 \leq t_i \leq 10^9$  trong tất cả các test
- 20% với  $1 \leq n \leq 5000$
- 20% với  $Ll_i = 1$ ,  $Lr_i = i$  với mọi  $i = 1, 2, \dots, n$

11/11:

## Bài E. FBRA

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

Từ thời xa xưa, dãy ngoặc được xem như là biểu tượng của tình yêu đôi lứa. Với Hoang\_Thuy, nó còn là lần đầu anh gặp Thuy\_Hoang. Một cách tự nhiên như cách con người tìm ra lửa, anh ta trở nên yêu quý các dãy ngoặc đúng:

- Xâu rỗng là một dãy ngoặc đúng
- Nếu A là dãy ngoặc đúng thì (A),[A],{A} cũng là dãy ngoặc đúng
- Nếu A,B là 2 dãy ngoặc đúng thì AB cũng là dãy ngoặc đúng

Mọi việc đều êm đềm cho đến một ngày, những dãy ngoặc có vẻ đúng của Hoang\_Thuy viết tăng cho Thuy\_Hoang bị mờ đi một vài chỗ. Bằng cách điền các dấu ngoặc vào những chỗ bị mờ, cô nàng muốn dãy ngoặc thu được là đúng. Tuy nhiên, cô tỏ ra khá bối rối vì có thể có nhiều cách điền khác nhau. Trong phút yêu lòng, cô muốn biết liệu có bao nhiêu cách điền, và cô sẵn sàng trả ơn cho ai tính được

### Dữ liệu vào

Gồm một dòng duy nhất chứa một xâu  $S$  chỉ gồm các ký tự thuộc {”, ”, ’[, ’], ’(, ’), ’?’} mô tả lại dãy ngoặc, các ký tự ’?’ mô tả các vị trí bị mờ

### Kết quả

Một số nguyên duy nhất là số dư của số cách điền vào những chỗ bị mờ khi chia cho  $10^9 + 7$ . Hai cách điền được coi là khác nhau nếu dãy ngoặc thu được từ chúng khác nhau

### Ví dụ

stdin	stdout
(? [??)	4
( ) [ ( ) ] { }	1
)????	0

### Hạn chế

- 30% test có  $|S| \leq 10$
- 70% test có  $10 < |S| \leq 500$

## Bài F. MNUM

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

Trên hành tinh XYZ, việc quay thưởng xổ số được làm như sau: Kết quả xổ số là 1 số có  $k$  chữ số. Họ tiến hành quay từng số bằng cách:

- Chọn 1 số nguyên dương  $n$
- Viết các số tự nhiên từ 1 đến  $n$  liền nhau thu được xâu  $S$
- Nếu độ dài  $S$  là 1 thì  $S$  chính là kết quả của lần quay
- Nếu độ dài  $S$  lớn hơn 1, xóa hết các ký tự ở vị trí lẻ của  $S$ , các ký tự còn lại được viết lại liền nhau
- Xóa hết các ký tự ở vị trí chẵn của  $S$ , các ký tự còn lại được viết lại liền nhau, quay lại bước 3

Cho biết thông tin chọn số của các lần quay, hãy tính toán và đưa ra kết quả xổ số. Lưu ý, hành tinh này sử dụng hệ đếm 10, và các ký tự trên xâu  $S$  được đánh số từ trái sang phải bắt đầu bằng 1

### Dữ liệu vào

- Dòng đầu chứa số nguyên dương  $k$
- Dòng tiếp theo chứa  $k$  số nguyên dương là số được chọn trong các lần quay

### Kết quả

- Kết quả xổ số ở hành tinh XYZ. Gồm  $k$  chữ số viết liền nhau

### Ví dụ

stdin	stdout
2	26
5 6	

### Hạn chế

- $1 \leq k \leq 10^5$ ,  $1 \leq n \leq 10^{15}$
- Subtask 1:  $k, n \leq 10$
- Subtask 2:  $k, n \leq 100$
- Subtask 3:  $k, n \leq 1000$
- Subtask 4:  $k, n \leq 100000$
- Subtask 5: Ràng buộc gốc

## Bài G. INTERMIN

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

Nhung đang giải một bài interactive như sau:

- Cần đoán một số nguyên  $x$  biết rằng số này thuộc từ 1 đến  $n$
- Mỗi lần có thể hỏi hệ thống là  $x$  có nhỏ hơn  $k$  hay không, với  $k$  là một số thực tùy ý
- Nếu hệ thống trả lời là “có”, Nhung mất chi phí  $a$  đồng. Ngược lại thì mất  $b$  đồng

Vì không quen với dạng bài interactive, Nhung đem bài toán hỏi Như - chị gái mình. Là một khuê nữ, Như muốn biết với  $n$  đầu vào, cần chi phí ít nhất là bao nhiêu để chắc chắn sẽ tìm ra được số  $x$ . Tất nhiên rồi, cô đem bài này hỏi Thái...

### Dữ liệu vào

- Dòng đầu chứa số test:  $Q$
- $Q$  dòng tiếp theo, mỗi dòng ghi một test:  $n \ a \ b$

### Kết quả

Gồm  $Q$  dòng ghi kết quả tương ứng

### Ví dụ

stdin	stdout
3	4
10 1 1	7
20 1 2	8
30 2 1	

### Hạn chế

- $1 \leq Q \leq 100$ ,  $1 \leq n, a, b \leq 10^5$
- Có ít nhất 50% số test với:  $1 \leq n \leq 100$

## Bài H. DGRAPH

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 ( $s_a \neq t_a$ ).

Do tính chất của mạng quang không lọc, thông tin được gửi đi từ một nút nào đó sẽ lan truyền khắp nơi. Các đỉnh nhận sẽ phân biệt gói tin dựa vào chiều truyền đến của gói tin và bước sóng của kết nối. Ta nói kết nối  $a$  làm nhiễu kết nối  $b$  nếu tin từ  $s_a$  và từ  $s_b$  đến  $t_b$  theo cùng một chiều, cụ thể là  $\text{Path}(s_a, t_b)$  và  $\text{Path}(s_b, t_b)$  có cạnh chung.

Trên thực tế, việc gán bước sóng cho các kết nối sẽ đưa về bài toán tô màu trên đồ thị quan hệ "làm nhiễu" nói trên. Tuy nhiên trong bài này, bạn chỉ cần tính số cung của đồ thị quan hệ đó, tức là số cặp  $a, b$  mà  $a$  làm nhiễu  $b$ .

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 cặp  $a, b$  mà  $a$  làm nhiễu  $b$ .

### 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	0
1 2	1
1 3	4
2 4	8
2 5	3
3 6	
3 7	
0 2 7	
0 3 1	
0 1 4	
0 4 6	
1 2 7	

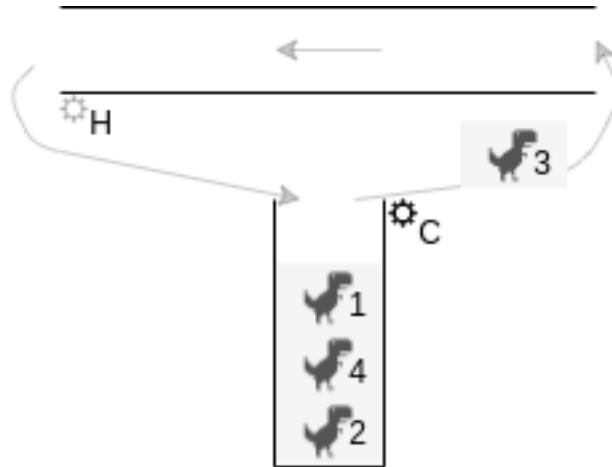
### Hạn chế

- $1 \leq n, m \leq 10^5$
- 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 I. SDINO

File dữ liệu vào: **stdin**  
File kết quả: **stdout**  
Hạn chế thời gian: 1.5 giây  
Hạn chế bộ nhớ: 512MB

Khủng long kỳ thực không hề hung dữ như người ta vẫn tưởng, chúng rất thông minh và dễ bảo. Bạn là người trông giữ khủng long và có nhiệm vụ sắp xếp lại khủng long trong chuồng. Chuồng chỉ có một cửa ra vào, và chiều ngang hẹp chỉ vừa đủ cho một con khủng long di chuyển, vì thế con nào vào chuồng trước thì sẽ phải ra sau. Ở cạnh chuồng có một hành lang. Hành lang có một cửa vào và một cửa ra, và chiều ngang cũng vừa đủ cho một con khủng long di chuyển, nên con nào vào trước sẽ phải ra trước.



Có  $n$  con khủng long, tất cả đều đang ở trong chuồng. Con khủng long thứ  $i$  (tính từ cửa chuồng vào trong) có số hiệu  $p_i$  ( $p_1, p_2, \dots, p_n$  là một hoán vị của  $\{1, 2, \dots, n\}$ ). Để sắp xếp lại các con khủng long, bạn sẽ sử dụng đèn báo hiệu được lắp ở chuồng và hành lang. Khi bật đèn báo hiệu ở chuồng, nếu trong chuồng có khủng long thì một con khủng long trong chuồng sẽ đi sang hành lang. Khi bật đèn báo hiệu ở hành lang, nếu ở hành lang có khủng long thì một con khủng long ở hành lang sẽ đi vào chuồng.

**Yêu cầu:** Hãy chỉ ra một dãy không quá  $10^6$  báo hiệu để sắp xếp lại các con khủng long theo thứ tự từ nhỏ đến lớn tính từ cửa chuồng vào trong.

### Dữ liệu vào

- Dòng đầu chứa số nguyên dương  $n$
- Dòng tiếp theo chứa  $n$  số nguyên dương  $p_1 \ p_2 \dots \ p_n$

### Kết quả

Ghi xâu  $s$  gồm nhiều ký tự viết liền nhau, các ký tự  $C$  cho biết bạn bật đèn ở chuồng, các ký tự  $H$  cho biết bạn bật đèn ở hành lang. Lưu ý các đèn là đèn báo hiệu và sẽ tắt ngay sau khi được bật sáng. Các đèn được bật theo đúng trình tự trong xâu  $s$ . Nếu  $s$  là một dãy điều khiển thỏa mãn, bạn sẽ nhận được  $\min(1, \frac{10^6}{|s|})$  tổng điểm của test đó.

### Ví dụ

stdin	stdout
4	CCHHCCCCHHHHCCCHCCCHCCCHHH
3 1 4 2	

### Hạn chế

- Có 12% số test với  $n \leq 50$ ;
- Có 24% số test với  $n \leq 1000$ ;
- Có 64% số test với  $n \leq 20000$ .

## Bài J. HATE

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

Một lớp học có  $n$  học sinh được đánh số từ 1 đến  $n$ , học sinh thứ  $i$  có năng lực là  $a_i$ . Sắp tới thầy chủ nhiệm muốn chọn ra một đội để tham gia thi đấu.

Tất nhiên ở trong lớp có thể có những bạn ghét nhau. Để tránh mất đoàn kết khi thi đấu, các học sinh sẽ cho thầy biết người mà mình ghét nhất (có thể bạn ghét nhiều người nhưng đừng đòi hỏi cao, thầy chỉ quan tâm người mà bạn ghét nhất thôi!!). Gọi  $b_i$  là người mà học sinh thứ  $i$  ghét nhất ( $b_i = -1$  có nghĩa là  $i$  không ghét ai cả). Thầy chủ nhiệm sẽ phải chọn đội sao cho nếu  $i$  được chọn thì  $b_i$  sẽ không được chọn, và tổng năng lực của đội được chọn là lớn nhất có thể.

Hãy giúp thầy!

### Dữ liệu vào

- Dòng đầu chứa số testcase:  $T$
- Mỗi testcass, dòng đầu tiên:  $n$
- $n$  dòng tiếp theo, dòng thứ  $i$  chứa:  $a_i$   $b_i$

### Kết quả

Ghi  $T$  dòng là tổng năng lực lớn nhất có thể của đội được chọn tương ứng cho  $T$  testcase

### Ví dụ

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

### Hạn chế

- $T \leq 10$  trong tất cả các test
- 50% số test với  $n \leq 20$
- 30% số test với  $20 < n \leq 1000$
- 20% số test với  $1000 < n \leq 10^5$

## Bài K. 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 L. VIRUS

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

Hùng đang phát triển một phần mềm diệt virus dựa trên nguyên tắc so khớp mẫu. Theo đó, cậu có tập  $S$  các xâu nhị phân là mẫu thường gặp trong mã nguồn của virus (mã độc). Mỗi tệp tin  $f$  có thể được hiểu như một xâu nhị phân, độ tương thích của  $f$  với mẫu  $x$  là số lần xuất hiện của  $x$  ở trong  $f$  (xuất hiện theo nghĩa bằng với một đoạn con liên tiếp). Để đánh giá  $f$  có bị nhiễm virus hay không, Hùng muốn tính tổng độ tương thích của  $f$  với mọi xâu trong  $S$ .

Trong quá trình sử dụng, Hùng có thể cập nhật thêm các mẫu virus mới. Ban đầu Hùng chưa có mẫu virus nào được báo cáo. Cụ thể bạn cần thực hiện hai loại truy vấn sau:

- Loại một: 0  $x$ : Thêm xâu  $x$  vào tập  $S$ . Nếu  $x$  đã xuất hiện trong  $S$ , Hùng vẫn thêm  $x$  vào (theo cậu, càng có nhiều người dùng báo cáo một mẫu thì mẫu đó càng quan trọng). Có thể hiểu  $S$  là một multiset.
- Loại hai: 1  $f$ : Tính toán và đưa ra tổng số lần xuất hiện của các xâu trong  $S$  trên xâu  $f$ .

### Dữ liệu vào

- Dòng đầu tiên chứa số nguyên dương  $Q$
- $Q$  dòng tiếp theo mỗi dòng mô tả một truy vấn: 0  $x$  hoặc 1  $f$

### Kết quả

- Với mỗi truy vấn loại hai, in ra kết quả trên một dòng

### Ví dụ

stdin	stdout
5	
0 0	10
0 1	
0 01	11
1 10011001	
1 01010011	

### Hạn chế

- $1 \leq Q \leq 10^6$ , tổng độ dài tất cả các xâu đầu vào không quá  $2 \times 10^6$
- Subtask 0:  $|S| \leq 10$
- Subtask 1: Các truy vấn loại hai nằm liên tiếp nhau
- Subtask 2: Ràng buộc gốc

18/11:

## Bài M. ZFU

File dữ liệu vào: **stdin**

File kết quả: **stdout**

Hạn chế thời gian: 1 giây

Hệ thống bảo mật của quân đội XYZ gồm một dãy  $n$  khóa (gọi là dãy  $S$ ), các khóa được đánh số từ 1 đến  $|S|$ . Có thể có nhiều loại khóa khác nhau, mỗi loại đặc trưng bởi một ký tự khác nhau. Để phá hủy hệ thống bảo mật đó, quân đội nhân dân ABC đã chế tạo một robot. Robot này sẽ phá lần lượt từng loại khóa theo thứ tự đã được lập trình sẵn, và nếu loại khóa cần phá hủy khác với loại phá khóa đang được sử dụng theo lệnh lập trình, robot sẽ không thể phá được khóa đó và dừng toàn bộ lệnh sau đó.

Ban đầu, robot được lập trình theo đúng dãy  $S$ . Dĩ nhiên nó sẽ phá được hết các khóa nếu được đặt vào vị trí đầu tiên của  $S$ , tuy nhiên việc tiếp cận dãy khóa này không hề dễ dàng. Theo đó, chim ruồi (một loại máy bay không người lái) sẽ mang robot và thả xuống khu vực có dãy khóa. Nếu bay đến vị trí  $i$  trên dãy khóa, chim ruồi có thể thả robot xuống các vị trí  $j \geq i$ , tức các vị trí từ  $i$  trở về sau của dãy. Tất nhiên là  $i$  càng nhỏ sẽ càng có cơ may phá được nhiều khóa hơn, tuy nhiên mức độ nguy hiểm cũng cao hơn. Để cân bằng giữa rủi ro và kết quả, quân đội nhân dân ABC muốn trả lời  $Q$  câu hỏi, mỗi câu hỏi có dạng: "Nếu chim ruồi bay đến vị trí  $i$  thì có nhiêu nhất bao nhiêu khóa sẽ bị phá".

### Dữ liệu vào

- Dòng đầu tiên chứa một xâu ký tự mô tả cho dãy khóa  $S$
- Dòng tiếp theo chứa số nguyên dương  $Q$
- $Q$  dòng tiếp theo, mỗi dòng chứa một số  $i$

### Kết quả

- Ghi ra  $Q$  dòng, mỗi dòng ghi kết quả cho một câu hỏi tương ứng

### Ví dụ

stdin	stdout
abaababaab	10
3	5
1	2
2	
8	

### Hạn chế

- $1 \leq |S|, Q \leq 10^6$
- Có 30% số test với  $1 \leq |S|, Q \leq 500$
- Có 30% số test với  $501 \leq |S|, Q \leq 5000$

## Bài N. SAG

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

Một xâu  $S$  độ dài  $n$  chỉ gồm các ký tự trong số  $k$  chữ cái đầu của bảng chữ cái latin thường ( $1 \leq k \leq 26$ ). Một người đã tiến hành ghi chép các thông tin về xâu này. Mỗi lần, anh ta chọn 2 vị trí  $i, j$  và tính ra ký tự  $c$  là  $=, >$ ,  $<$  tương ứng là  $S_i = S_j$ ,  $S_i > S_j$ ,  $S_i < S_j$

Cho biết  $m$  thông tin dạng trên, hãy khôi phục lại xâu  $S$

### Dữ liệu vào

- Dòng đầu:  $n \ k \ m$
- $m$  dòng tiếp theo, mỗi dòng ghi một thông tin:  $i \ c \ j$  (viết liền nhau)

Chỉ số của xâu đánh số từ 1, dữ liệu đảm bảo không mâu thuẫn

### Kết quả

- Ghi xâu  $S$ . Với các ký tự không thể xác định được, ghi ?

### Ví dụ

stdin	stdout
4 3 2 1<2 2<3	abc?
5 3 3 1>2 2>3 5=2	cba?b

### Hạn chế

- Subtask 1:  $1 \leq n, m \leq 1000$
- Subtask 2:  $1 \leq n, m \leq 10^5$

## Bài O. SHOOT2D

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

Bạn có bao giờ thắc mắc, đi nghĩa vụ quân sự là như thế nào? Sau đây là ghi chép về một buổi tập trận của các thanh niên đó:

Vùng ngắm bắn có thể được hiểu như trực tọa độ Oxy, với trục Ox ở mặt đất. Có  $n$  sự kiện diễn ra trong buổi tập, được đánh số từ 1 đến  $n$ . Mỗi sự kiện thuộc một trong hai dạng sau:

- 1  $x$   $y$ : Chỉ huy ra lệnh đặt một tấm bia có bán kính  $y$  vào vị trí  $x$  (tức là tâm của bia trùng vào điểm  $(x, y)$ )
- 2  $x$   $y$ : Một thanh niên bắn vào điểm  $(x, y)$

Việc đặt bia đảm bảo luôn không có hai bia nào đè lên nhau (có thể tiếp xúc). Đồng thời, mỗi lần bắn trúng thì bia bị trúng đó sẽ biến mất. Để tính điểm cho các thanh niên vừa nhập ngũ, chỉ huy cần biết mỗi phát bắn sẽ bắn trúng bia nào, hoặc thông báo là thanh niên này bắn trượt (chạm vào biên của bia không được tính là bắn trúng)

### Dữ liệu vào

- Dòng đầu tiên chứa số nguyên:  $n$
- $n$  dòng tiếp theo mỗi dòng ghi ba số nguyên mô tả một sự kiện:  $t$   $x$   $y$

### Kết quả

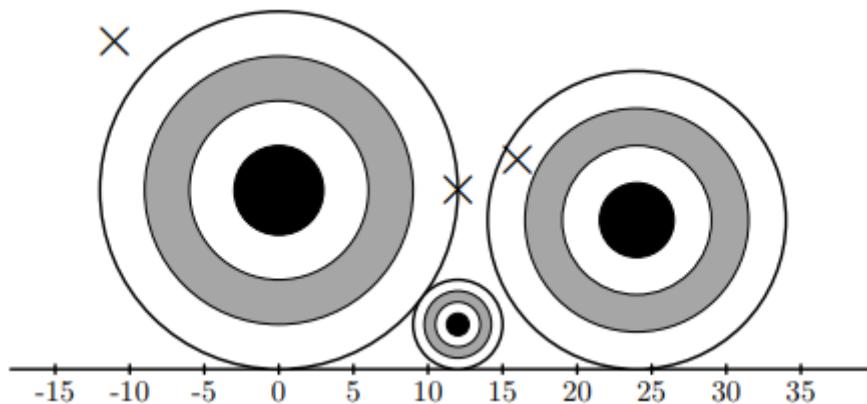
- Với mỗi sự kiện loại 2, in ra chỉ số của sự kiện đặt bia bị bắn trúng, hoặc -1 nếu phát bắn không trúng bất cứ bia nào

### Ví dụ

stdin	stdout
8	-1
1 0 12	-1
2 -11 22	3
1 24 10	1
1 12 3	
2 12 12	
2 16 14	
1 28 15	
2 3 6	

### Giải thích

Hình sau giải thích cho test ví dụ:



## Hạn chế

- $1 \leq n \leq 2 \times 10^5$ ,  $|x|, |y| \leq 10^9$ ,  $y > 0$

## Bài P. SUMLOVE

File dữ liệu vào: **stdin**  
File kết quả: **stdout**  
Hạn chế thời gian: 3 giây

Tương truyền, Ngưu Lang và Chức Nữ yêu nhau thắm thiết đến độ có thể thần giao cách cảm. Để kiểm chứng điều này, Ngọc Hoàng đã đưa cho họ một số nguyên dương  $n$ . Hàng năm vào ngày rằm tháng bảy, Ngọc Hoàng sẽ đưa cho họ thêm một số nguyên dương  $k$ , sau đó bảo hai người họ mỗi người chọn một tập con của  $\{1, 2, \dots, n\}$ . Nếu tổng các số Ngưu Lang chọn cộng với tổng các số Chức Nữ chọn bằng đúng  $k$  thì họ sẽ được gặp nhau.

**Yêu cầu:** Hãy giúp cặp đôi này tính toán số cách chọn khác nhau cho từng năm để họ được gặp nhau. Hai cách chọn được cho là khác nhau nếu tập mà Ngưu Lang chọn là khác nhau trong hai cách đó hoặc tập mà Chức Nữ chọn là khác nhau trong hai cách đó

### Dữ liệu vào

- Dòng đầu chứa hai số nguyên dương  $n$   $Q$  với  $Q$  là số năm
- Mỗi dòng trong  $Q$  dòng tiếp theo ghi một số nguyên dương  $k$

### Kết quả

Ghi  $Q$  dòng là số cách chọn tương ứng cho  $Q$  năm, sau khi chia lấy dư cho  $10^9 + 7$

### Ví dụ

stdin	stdout
5 5	2
1	3
2	6
3	9
4	14
5	

### Hạn chế

- Trong tất cả các test:  $n, k, Q \leq 10^5$
- Có 25% số test với  $k \leq 500$
- Có 25% số test với  $n \leq 500$
- Có 25% số test với  $Q \leq 500$
- Có 25% số test với ràng buộc gốc