

01/02:

Bài A. RTREE2

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

Mạng lưới thông tin trên hành tinh XYZ có thể được hiểu như một cây n đỉnh (một đồ thị liên thông không có chu trình), mỗi đỉnh là một trạm, còn mỗi cạnh là một đường truyền. Các trạm được đánh số từ 1 đến n , các đường truyền được đánh số từ 1 đến $n - 1$.

Mỗi kết nối trên mạng là 1 luồng dữ liệu đang được truyền tải, thể hiện bởi số hiệu của 2 trạm gửi-nhận. Cho m sự kiện theo thứ tự, mỗi sự kiện thuộc 1 trong 2 dạng sau:

- 1 $u\ v$: Có thêm một kết nối giữa u và v
- 2 i : Quân đội muốn tính số kết nối bị gián đoạn nếu cắt cạnh thứ i

Dữ liệu vào

- Dòng đầu: $n\ m$
- $n - 1$ dòng tiếp theo ghi các cạnh của cây: $u\ v$
- m dòng tiếp theo ghi các sự kiện theo mô tả ở trên

Kết quả

- Với mỗi sự kiện loại 2, in ra kết quả trên một dòng. Lưu ý là chúng ta chỉ tính toán chứ chưa thực sự phá hủy đường truyền nào

Ví dụ

stdin	stdout
3 3 1 2 1 3 1 2 3 1 2 1 2 1	2

Hạn chế

- $1 \leq n, m \leq 10^5$
- 50% số test có $1 \leq n, m \leq 5000$

Bài B. SMEX2

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

Cho $a = a_1, a_2, \dots, a_n$ là một dãy số nguyên không âm. Hãy đếm số cách chia dãy a thành các đoạn con liên tiếp, sao cho MEX của các đoạn con, theo thứ tự, tạo thành một dãy không giảm. Ở đây, MEX của đoạn con $[i..j]$ là số nguyên không âm nhỏ nhất không xuất hiện trong $\{a_i, a_{i+1}, \dots, a_j\}$.

Dữ liệu vào

- Dòng đầu chứa: n ;
- Dòng tiếp theo chứa: $a_1 \ a_2 \ \dots \ a_n$

Kết quả

Ghi số cách chia tìm được theo modulo 1000000007.

Ví dụ

stdin	stdout
8 3 0 2 1 0 1 3 2	8

Hạn chế

- Trong tất cả các test: $1 \leq n \leq 5000$; $0 \leq a_i \leq 10^9$;
- Có 24% số test với $n \leq 100$;
- Có 32% số test với $a_i \leq 100$;
- Có 44% số test với ràng buộc gốc.

Bài C. DOPA

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

Một xâu được gọi là đối xứng kép nếu nó có thể tách làm hai xâu đối xứng (một trong hai xâu đó có thể rỗng). Nói cách khác, xâu đối xứng kép là xâu đối xứng hoặc được ghép từ hai xâu đối xứng. Ví dụ xâu banana, abcba là các xâu đối xứng kép. Hãy đếm số xâu đối xứng kép độ dài không quá n , chỉ chứa các ký tự latin thường

Dữ liệu vào

- Dòng duy nhất chứa số nguyên dương n

Kết quả

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

Ví dụ

stdin	stdout
3	2678
4	38454

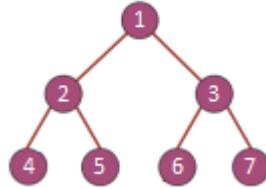
Hạn chế

- $1 \leq n \leq 10^5$
- 20% test có $n \leq 15$
- 20% test có $n \leq 1000$

Bài D. CXTREE

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

Cây là một đơn đồ thị liên thông không có chu trình, mỗi đỉnh có một nhãn khác nhau. Tải trọng của một cạnh là số cặp đỉnh mà đường đi đơn giữa chúng phải đi qua cạnh đó (cặp (x, y) và cặp (y, x) được coi là giống nhau)



Hãy đếm số lượng cây có n đỉnh và tải trọng của các cạnh không vượt quá k . Hai cây được coi là khác nhau nếu tồn tại một cặp đỉnh mà trên cây này thì có cạnh nối trực tiếp, còn trên cây kia thì không

Dữ liệu vào

Ghi hai số n k

Kết quả

Một số nguyên là kết quả bài toán lấy dư khi chia $10^9 + 7$

Ví dụ

stdin	stdout
3 2	3

Hạn chế

- $1 \leq n \leq 5000, n - 1 \leq k \leq n(n - 1)/2$
- 30% test: $1 \leq n \leq 13$
- 30% test: $14 \leq n \leq 100$

Bài E. FCANDLE

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

Những ngon nển lung linh huyền ảo khiến biết bao người đam mê, trong số đó có Nhan_Tai. Một hôm, Tai_Nhan đưa cho anh n cây nến với độ cao đôi một khác nhau. Theo đó, Nhan_Tai sẽ phải xếp n cây nến này theo một đường thẳng sao cho các cây nến ở vị trí chẵn thì cao hơn hai cây nến hai bên (nếu có), và dĩ nhiên, các cây nến ở vị trí lẻ thì thấp hơn hai cây nến hai bên (nếu có). Cụ thể, cách xếp phải thỏa mãn với mọi $1 \leq i \leq n$:

- Nếu i chẵn: $i > 1 \Rightarrow h_{i-1} < h_i$; $i < n \Rightarrow h_{i+1} < h_i$
- Nếu i lẻ: $i > 1 \Rightarrow h_{i-1} > h_i$; $i < n \Rightarrow h_{i+1} > h_i$

Ở đây h_i được hiểu như là độ cao của cây nến xếp ở vị trí thứ i trên đường thẳng. Anh ta muốn biết mình có bao nhiêu cách khác nhau để xếp các cây nến. Vốn nhìn xa trông rộng, Nhan_Tai biết sẽ có thể có rất nhiều cách xếp, anh ta chỉ yêu cầu in ra 9 chữ số tận cùng của số cách xếp

Dữ liệu vào

- Dòng đầu chứa 2 số nguyên Q là số lượng testcase
- Mỗi testcase nằm trên một dòng chứa đúng một số nguyên dương: n

Kết quả

Gồm Q dòng trả lời cho Q testcase

Ví dụ

stdin	stdout
5	000000061
6	000000272
7	000001385
8	000007936
9	000050521
10	

Hạn chế

- $N, Q \leq 5000$
- Có 50% số test với $n \leq 100$

Bài F. RTREE3

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

Chúng ta đều biết thuyết tiến hóa của Darwin là một học thuyết có tính khoa học, hệ thống, có nhiều bằng chứng thực nghiệm. Ấy vậy mà gần đây có không ít người, có thể là bao gồm những học sinh không thích môn sinh học, tỏ ra không tin tưởng và thậm chí phản bác nó một cách vô căn cứ.

Cả hai phe, ủng hộ và không ủng hộ, mỗi bên đều đưa ra một mô hình cây tiến hóa, trong đó các loài được đánh số từ 1 đến n . Nếu x là tổ tiên của y thì ở trên cây, x sẽ nằm trên đường đi đơn từ y đến 1. Hai cây này cũng có những điểm giống nhau, chẳng hạn cả hai đều công nhận 1 là tổ tiên của tất cả các loài. Để đánh giá mức độ giống nhau của hai mô hình, với mỗi y , cần tính số lượng x là tổ tiên của y ở cả hai mô hình đó.

Dữ liệu vào

- Dòng đầu chứa số nguyên dương: n ; ($1 \leq n \leq 5 \times 10^5$)
- $n - 1$ dòng tiếp theo, mỗi dòng chứa hai số tự nhiên mô tả một cạnh của cây thứ nhất: x, y ;
- $n - 1$ dòng tiếp theo, mỗi dòng chứa hai số tự nhiên mô tả một cạnh của cây thứ hai: x, y ;

Kết quả

Ghi ra n số, số thứ y là số lượng x là tổ tiên của y ở cả hai mô hình.

Ví dụ

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

Bài G. TWOGRAPH

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

Kiến thường sống thành từng đê chế với tổ chức xã hội rất cao. Có hai đê chế kiến, đỏ và đen. Mỗi đê chế kiến được mô tả bởi một đồ thị vô hướng có trọng số. Do chỉ có một cáp viễn thông giữa hai đê chế, để truyền tải dữ liệu giữa thành phố của hai đê chế trước tiên dây cáp phải được di chuyển đến hai thành phố tương ứng. Giả sử hiện tại dây cáp đang nối thành phố u -đỏ và v -đen, ta nói dây cáp đang ở (u, v) . Việc chuyển cáp có thể gồm nhiều bước, mỗi bước thuộc một trong hai loại sau:

- Chuyển dây cáp từ (u, v) sang (x, v) . Việc này phải sử dụng một số tuyến đường của đê chế đỏ để đi từ u sang x . Tất cả các tuyến đường sử dụng đến đều phải đã được đóng phí bởi thành phố v
- Chuyển dây cáp từ (u, v) sang (u, y) . Việc này phải sử dụng một số tuyến đường của đê chế đen để đi từ v sang y . Tất cả các tuyến đường sử dụng đến đều phải đã được đóng phí bởi thành phố u

Mỗi thành phố của đê chế này cần phải chọn ra một số tuyến đường của đê chế kia để đóng phí, sao cho hệ thống đảm bảo mọi cặp (u, v) đều có thể được truyền tải dữ liệu với nhau. Lưu ý là khi một thành phố đóng phí sử dụng cho một tuyến đường nào đó, họ chỉ đóng phí một lần và được sử dụng tuyến đường đó nhiều lần, mức phí của mỗi tuyến đường là như nhau với mọi thành phố. Tuy nhiên việc thu phí của một tuyến đường có phân biệt các thành phố, tức là mỗi tuyến đường có thể sẽ được đóng phí bởi nhiều thành phố khác nhau. Hãy tính tổng phí nhỏ nhất có thể mà tất cả các thành phố phải đóng

Dữ liệu vào

Đồ thị mô tả hai đê chế sẽ được đọc vào lần lượt, mỗi đồ thị gồm có:

- Dòng đầu tiên chứa n m là số đỉnh và số cạnh
- m dòng tiếp theo mỗi dòng chứa u v w là thành phố hai đầu một tuyến đường và chi phí phải đóng để sử dụng tuyến đường đó

Kết quả

- Ghi ra tổng phí nhỏ nhất tìm được. Nếu không thể truyền tải dữ liệu cho mọi cặp, in ra -1

Ví dụ

stdin	stdout
6 9 1 4 52 2 1 100 1 3 94 6 4 9 6 3 97 5 6 17 2 6 35 6 3 15 2 3 37 8 7 3 2 90 1 6 68 5 8 79 7 3 71 1 3 47 7 5 15 6 4 30	1333

Hạn chế

- $1 \leq n, m \leq 10^5, 1 \leq w \leq 10^9$
- Có 50% số test với $1 \leq n, m \leq 1000$

Bài H. TSELL

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

Câu lạc bộ khởi nghiệp vừa mang khẩu trang đến một đất nước tươi đẹp để bán. Đất nước này có dạng một cây n đỉnh tương ứng với n thành phố. Thành phố thứ i có khả năng mua w_i khẩu trang. Câu lạc bộ gồm k người, người thứ i được phép bán hàng cho các thành phố trên đường đi đơn từ a_i đến b_i , và anh ta mang theo c_i khẩu trang. Hỏi có thể bán được tổng nhiều nhất là bao nhiêu khẩu trang

Dữ liệu vào

- Dòng đầu ghi hai số nguyên dương: n k ($1 \leq n, k \leq 10^4$)
- Dòng tiếp theo chứa: w_1 w_2 \dots w_n ($0 \leq w_i \leq 10^5$)
- $n - 1$ dòng tiếp theo, mỗi dòng chứa một cạnh của cây: u v
- k dòng cuối, dòng thứ i ghi: a_i b_i c_i ($1 \leq c_i \leq 10^5$)

Kết quả

Ghi một số nguyên duy nhất là số khẩu trang tối đa bán được

Ví dụ

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

01/02:

Bài I. CXPAIR

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

Cho đơn đồ thị vô hướng không có khuyên. Đếm số cách thêm vào một cạnh để số chu trình đơn của đồ thị tăng lên đúng một.

Nhắc lại, chu trình đơn là một dãy các đỉnh khác nhau đôi một x_1, x_2, \dots, x_k sao cho $k \geq 3$ và x_1 kề với x_k , x_i kề với x_{i+1} với mọi $1 \leq i < k$. Các chu trình là xoay vòng của nhau được xem là giống nhau.

Dữ liệu vào

- Dòng đầu chứa: n m là số đỉnh và số cạnh, các đỉnh sẽ được đánh số từ 1 đến n
- m dòng tiếp theo, mỗi dòng chứa một cạnh: x y

Kết quả

Chứa số nguyên duy nhất là kết quả bài toán

Ví dụ

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

Hạn chế

- $1 \leq n, m \leq 10^5$
- 50% với $2 \leq n \leq 5000$

Bài J. RTREE5

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

Đất nước XYZ có n ngôi làng và $n - 1$ con đường đảm bảo đi lại giữa mọi ngôi làng, nói cách khác, đất nước có dạng một cây. Chính phủ muốn xây dựng nông thôn mới ở tất cả các ngôi làng. Chi phí để xây dựng nông thôn mới ở ngôi làng thứ i là a_i . Chi phí này sẽ do chính phủ chi trả nếu địa phương không chủ động xây dựng. Hoặc sẽ do địa phương tự chi trả nếu họ chủ động xây dựng. Ngôi làng thứ i sẽ chủ động xây dựng nếu có ít nhất b_i ngôi làng kề cạnh với i đã thực hiện xây dựng. Hãy tính tổng chi phí nhỏ nhất chính phủ cần bỏ ra để xây dựng nông thôn mới ở tất cả các ngôi làng. Thời gian cần thiết để xây dựng là không quan trọng, tức là chính phủ có thể chờ bao lâu cũng được.

Dữ liệu vào

- Dòng đầu chứa số nguyên dương: n ;
- $n - 1$ dòng tiếp theo, mỗi dòng chứa hai số tự nhiên mô tả một cạnh của cây: u, v ;
- Dòng tiếp theo chứa n số nguyên: a_1, a_2, \dots, a_n ;
- Dòng tiếp theo chứa n số nguyên: b_1, b_2, \dots, b_n .

Kết quả

Ghi một số nguyên duy nhất là tổng chi phí ít nhất chính phủ cần bỏ ra.

Ví dụ

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

Hạn chế

- Trong tất cả các test: $1 \leq n \leq 10^5$; $1 \leq a_i, b_i \leq 10^9$;
- Có 24% số test với $n \leq 100$;
- Có 32% số test với mỗi đỉnh chỉ kề với không quá hai đỉnh khác;
- Có 44% số test với ràng buộc gốc.

Bài K. PFTREE

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

Cho một cây n đỉnh, các đỉnh được đánh số từ 1 đến n với gốc là đỉnh 1. Cây này là hoàn hảo, tức là khoảng cách từ gốc đến các nút lá đều bằng nhau. Trọng số của cây con gốc u tại thời điểm t , ký hiệu là $W(u, t)$, được tính như sau:

- Liệt kê khoảng cách từ u đến các nút trong cây con gốc u ;
- Sắp xếp tăng dần các khoảng cách đó. Gọi d_i là số thứ i trong dãy đã sắp xếp;
- Khi đó $W(u, t) = \sum_{i=1}^{sz(u)} d_i * t^{i-1}$ với $sz(u)$ là số đỉnh trong cây con gốc u .

Cho Q truy vấn, mỗi truy vấn là hai số u, t , cần tính $W(u, t) \% 1000000007$. Các truy vấn cần được trả lời online.

Dữ liệu vào

- Dòng đầu ghi: n Q ;
- $n - 1$ dòng tiếp theo, mỗi dòng ghi u, v là một cạnh của cây;
- Mỗi dòng trong số Q dòng tiếp theo ghi hai số a, b mô tả một truy vấn. Khi đó: $u = 1 + (a + p) \% n$, $t = 1 + (b + p) \% 1000000000$ với p là tổng kết quả của các truy vấn trước truy vấn này ($0 \leq a, b \leq 10^9$).

Kết quả

Với mỗi truy vấn, in ra trên một dòng là $W(u, t) \% 1000000007$.

Ví dụ

stdin	stdout
6 3	8
1 2	9
1 3	0
2 4	
2 5	
3 6	
0 0	
0 0	
0 0	

Hạn chế

- Trong tất cả các test: $1 \leq n, q \leq 10^5$;
- Có 8% số test với $n, q \leq 500$;
- Có 12% số test với $n, q \leq 5000$;
- Có 28% số test với mỗi đỉnh đều kề với nhiều nhất 2 đỉnh khác;
- Có 52% số test với ràng buộc gốc.

Bài L. WGCD

File dữ liệu vào:	<code>standard input</code>
File kết quả:	<code>standard output</code>
Hạn chế thời gian:	2 seconds
Hạn chế bộ nhớ:	256 megabytes

Trọng số của số nguyên dương x là $w(x)$. Cho lưới $m \times n$ gồm các số nguyên dương, số ở ô (i, j) ($1 \leq i \leq n, 1 \leq j \leq m$) là $\gcd(i, j)$ (ước chung lớn nhất của i và j). Hãy tính tổng trọng số của tất cả các số trên lưới.

Dữ liệu vào

- Dòng đầu chứa hai số nguyên dương: T H ($T, H \leq 10^5$) là số lượng trường hợp test và giới hạn của n, m trong T trường hợp test đó;
- Dòng thứ hai chứa H số nguyên là $w(1), w(2), \dots, w(H)$, các số có giá trị tuyệt đối không quá 10^8 ;
- Mỗi dòng trong T dòng tiếp theo chứa một trường hợp test bao gồm hai số nguyên dương: m n ($n, m \leq H$).

Kết quả

Ghi T dòng là kết quả cho T trường hợp test tương ứng theo thứ tự đầu vào.

Hạn chế

- Có 8% test ứng với $1 \leq H \leq 1000$;
- Có 12% test ứng với $1 \leq H \leq 10000$;
- Có 16% test ứng với $w(1) = 1, w(x) = 0 \forall x \neq 1$ và $n = m$;
- Có 20% test ứng với $w(1) = 1, w(x) = 0 \forall x \neq 1$ và $T \leq 10$;
- Có 20% test ứng với $w(p) = 1 \forall p$ là số nguyên tố, $w(x) = 0 \forall x$ là hợp số;
- Có 24% test còn lại với ràng buộc gốc.

Ví dụ

standard input	standard output
3 8	34
1 3 2 -5 3 2 3 1	7
4 8	76
7 1	
8 8	

Bài M. KDQUEEN

File dữ liệu vào: **stdin**
File kết quả: **stdout**
Hạn chế thời gian: 1 giây
Hạn chế bộ nhớ: 512MB

Cho một bàn cờ k chiều kích thước $n_1 \times n_2 \times \dots \times n_k$. Một quân hậu được đặt ở vị trí $X = (x_1, x_2, \dots, x_k)$, nó có thể di chuyển sang vị trí $Y = (y_1, y_2, \dots, y_k)$ ($1 \leq y_i \leq n_i$) nếu X và Y cùng hàng hoặc cùng đường chéo. Cùng hàng nghĩa là tồn tại i sao cho $|x_i \neq y_i|$ và $x_j = y_j \forall j \neq i$. Cùng đường chéo nghĩa là $|x_i - y_i| = |x_j - y_j| \forall 1 \leq i < j \leq k$. Hãy đếm số ô mà quân hậu có thể di chuyển sang (trong một nước đi).

Dữ liệu vào

- Dòng đầu ghi số nguyên dương k là số chiều của bàn cờ;
- Dòng thứ hai ghi k số nguyên dương n_1, n_2, \dots, n_k là kích thước của bàn cờ;
- Dòng thứ ba ghi k số nguyên dương x_1, x_2, \dots, x_k là vị trí của quân hậu.

Kết quả

Ghi số ô mà quân hậu có thể di chuyển sang sau khi chia lấy dư cho $10^9 + 7$

Ví dụ

stdin	stdout
3 3 3 3 1 2 3	8

Giải thích

Các ô đó là $(3, 2, 3); (2, 2, 3); (1, 1, 3); (1, 2, 1); (1, 3, 3); (1, 2, 2); (2, 1, 2); (2, 3, 2)$.

Hạn chế

- Trong tất cả các test: $2 \leq k \leq 10^5$, $2 \leq n_i \leq 10^9$;
- Có 20% số test với $k \leq 5$ và $n_i \leq 100$;
- Có 20% số test với $k \leq 1000$ và $n_i \leq 1000$;
- Có 20% số test với $n_i \leq 10^5$;
- Có 40% số test với ràng buộc gốc.

Bài N. KSET

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

Cho hai số nguyên dương n, k , đếm số tập S thỏa mãn:

- S chỉ chứa các số nguyên dương là ước của $n!$;
- $|S| = k$;
- Gọi P là tích các số thuộc S . Khi đó tất cả các ước nguyên tố của $n!$ đều là ước của P ;
- Hai phần tử khác nhau x, y bất kỳ của S đều thỏa mãn $N(x) * N(y) = N(x * y)$. Ở đây $N(m)$ là số ước nguyên dương của m .

Hai tập được coi là khác nhau nếu tồn tại một số thuộc tập này mà không thuộc tập kia.

Dữ liệu vào

- Dòng đầu chứa số lượng testcase: T ;
- T dòng tiếp theo, mỗi dòng chứa hai số: n, k .

Kết quả

Ghi T dòng là kết quả cho T testcase sau khi chia lấy dư cho $10^9 + 7$.

Ví dụ

<code>stdin</code>	<code>stdout</code>
2	32
6 3	182
8 3	

Hạn chế

- Trong tất cả các test: $1 \leq T, n, k \leq 10^6$; Tổng k trong tất cả các test không quá 10^6 ;
- Có 15% số test với $n \leq 1000$;
- Có 20% số test với $k = 2$;
- Có 25% số test với $1 \leq T \leq 10$;
- Có 40% số test với ràng buộc gốc.

Bài O. XAMBQ

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

Cho ma trận A có kích thước $n \times n$ chứa các giá trị trong khoảng 0 đến 1000. Tiến hành xây dựng ma trận B như sau:

Với (i, j) là một ô trên ma trận, xét mọi đường đi từ $A_{1,1}$ đến $A_{i,j}$ chỉ được đi sang phải và xuống dưới. Tiến hành tính tổng các số đi qua trên mỗi đường đi và chọn ra đường có tổng lớn nhất, gọi tổng đó là $B_{i,j}$. Một cách dễ hiểu hơn, $B_{i,j}$ là tổng trọng số lớn nhất có thể khi đi từ ô $(1, 1)$ đến ô (i, j) mà chỉ được đi sang phải (cột tăng một) hoặc xuống dưới (dòng tăng một).

Bài toán có vẻ đơn giản, nhưng A là một ma trận cà khịa. Thật vậy, A có thể thực hiện các biến đổi dạng UP($A_{x,y}$) hoặc DOWN($A_{x,y}$) tương ứng là tăng hoặc giảm $A_{x,y}$ đúng một đơn vị. Bạn cần tính lại ma trận B sau mỗi lần cà khịa như vậy. Tất nhiên là có quá nhiều số để in ra, vì vậy bạn chỉ cần in ra tổng của ma trận B sau khi tính lại mà thôi

Dữ liệu vào

- Dòng đầu tiên chứa số nguyên: n
- Tiếp theo là n dòng mỗi dòng n số nguyên mô tả A : $A_{x,y}$
- n dòng cuối ghi n thay đổi của A : $c \ x \ y$ với c bằng U hoặc D

Kết quả

- Gồm $n + 1$ dòng, dòng đầu tiên ghi tổng của ma trận B trước khi thay đổi ma trận A , mỗi dòng trong n dòng tiếp theo ghi tổng của ma trận B sau mỗi phép thay đổi của A

Ví dụ

XAMBQ.inp	XAMBQ.out
5	420
3 3 5 0 5	445
4 4 2 4 3	453
4 4 2 1 3	451
4 3 2 4 2	448
2 3 3 2 4	442
U 1 1	
U 1 3	
D 5 2	
D 4 3	
D 2 4	

Hạn chế

- $1 \leq n \leq 1500$
- Có 50% số test với $n \leq 500$
- Độ lớn của n sẽ được rải đều cho các test

Bài P. DTTUI4

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

Cho n đồ vật, vật thứ i có khối lượng c_i và giá trị v_i . Cho một cái túi có sức chứa S và một số tự nhiên k . Khi chọn ra một số đồ vật để cho vào túi, tổng khối lượng các đồ vật này phải không quá sức chứa của túi, khi đó giá trị của cách chọn là V^k với V là tổng giá trị các đồ vật được chọn. Hãy tính tổng giá trị của tất cả các cách chọn.

Dữ liệu vào

- Dòng đầu chứa n, S, k ;
- Dòng thứ i trong số n dòng tiếp theo chứa c_i, v_i .

Kết quả

Ghi tổng giá trị tìm được, sau khi chia lấy dư cho 998244353

Ví dụ

stdin	stdout
3 2 2 1 2 2 3 1 4	65

Hạn chế

- Trong tất cả các test: $1 \leq n, S, k, c_i \leq 100$; $1 \leq v_i \leq 10^9$
- Có 10% số test với $1 \leq N \leq 20$;
- Có 15% số test với $k = 0$;
- Có 15% số test với $k = 1$;
- Có 15% số test với $k = 2$;
- Có 45% số test với ràng buộc gốc.