

Dồn bò [flowers]

Bờm đi hái củi và để đàn bò gồm N con ăn cỏ trên cánh đồng như thường lệ. Khi quay lại, Bờm phát hiện đàn bò đã vào vườn hoa và đang chén những cây hoa. Bờm quyết định phải đưa lũ bò về chuồng ngay lập tức, hơn thế, còn phải dồn bò sao cho số hoa thiệt hại là ít nhất.

Con bò i đang ở địa điểm cách chuồng một khoảng T_i phút di chuyển, nó chén D_i cây hoa trong mỗi phút chờ tới lượt được đưa về chuồng. Ở mỗi thời điểm Bờm chỉ có thể đưa một con bò về chuồng, với con bò i Bờm sẽ mất $2 \times T_i$ phút (đưa bò về chuồng và quay lại).

Lập chương trình xác định số cây hoa bị ăn nếu Bờm dồn bò theo lịch tối ưu.

Dữ liệu (flowers.inp)

- Dòng 1: ghi số nguyên N ($2 \leq N \leq 100,000$)
- Dòng 2 ... $N + 1$: mỗi dòng ghi hai số nguyên T_i ($1 \leq T_i \leq 2,000,000$) và D_i ($1 \leq D_i \leq 100$).

Kết quả (flowers.out)

- Dòng 1: số nguyên là số cây hoa bị ăn ít nhất.

Ví dụ

flowers.inp	flowers.out
6 3 1 2 5 2 3 3 2 4 1 1 6	86

Xếp thùng [stack]

Cho N thùng chứa đồ đánh số $1 \dots N$, thùng i có trọng lượng W_i và có thể chịu được tổng trọng lượng không quá C_i đặt trên. Hãy xác định số thùng nhiều nhất có thể xếp được thành một chồng.

Dữ liệu (stack.inp)

- Dòng 1: số nguyên N ($1 \leq N \leq 2,500$)
- Dòng 2: N số nguyên W_1, W_2, \dots, W_N ($1 \leq W_i \leq 100,000$)
- Dòng 3: N số nguyên C_1, C_2, \dots, C_N ($1 \leq C_i \leq 1,000,000,000$)

Kết quả (stack.out)

- Dòng 1: số nguyên là số thùng nhiều nhất có thể xếp thành một chồng

Ví dụ

stack.inp	stack.out
3 10 20 30 11 100 10	3
3 11 20 30 11 100 10	2

Đoàn thám hiểm (expedi)

Một đàn bò lấy một chiếc xe tải và liền liền mở một cuộc thám hiểm vào sâu trong một khu rừng già. Thật không may, là những tài xế kém cỏi, đàn bò đã quyết định vượt qua một hòn đá bằng cách chạy vọt lên trên nó. Hậu quả là bình xăng của xe đã bị thủng. Chiếc xe bây giờ sẽ mất một đơn vị nhiên liệu cho mỗi đơn vị độ dài chúng phải đi.

Để sửa lại xe, những con bò phải đi đến thị trấn gần nhất trên một con đường dài và đầy gió. Trên con đường này, giữa thị trấn và vị trí hiện tại của xe có N trạm xăng. Chúng có thể dừng tại trạm xăng để đổ xăng.

Rừng già rất nguy hiểm cho con người, và càng đặc biệt nguy hiểm đối với bò. Do đó, chúng muốn dừng xe ít lần nhất trên đường đến thị trấn. Bình xăng của xe có kích thước coi như vô hạn. Chiếc xe hiện cách thị trấn L đơn vị độ dài và đang có sẵn P đơn vị xăng.

Xác định số lần ít nhất lũ bò phải dừng lại để có thể đến được thị trấn, hoặc chúng không thể thực hiện được việc này.

Dữ liệu (expedi.inp)

- Dòng 1: số nguyên N ($1 \leq N \leq 10,000$)
- Dòng 2 ... $N + 1$: mỗi dòng chứa hai số nguyên, được phân cách bởi khoảng trắng mô tả một trạm xăng. Số thứ nhất chỉ khoảng cách tính từ thị trấn, số thứ hai trong phạm vi 1 ... 100 chỉ lượng xăng có tại trạm đó.
- Dòng $N + 2$: hai số nguyên L, P ($1 \leq L, P \leq 1,000,000$).

Kết quả (expedi.out)

- Dòng 1: một số nguyên chỉ số trạm xăng cần phải ghé để có thể đến được thị trấn. Nếu không tồn tại khả năng này thì in ra -1 .

Ví dụ

expedi.inp	expedi.out
4 4 4 5 2 11 5 15 10 25 10	2

Giải thích

Ban đầu xe tải cách thị trấn 25 đơn vị độ dài, có 10 đơn vị xăng. Dọc con đường có 4 trạm xăng tại các mốc 4, 5, 11 và 15 tính từ thị trấn (tức là cách vị trí ban đầu của xe 21, 20, 14 và 10 đơn vị). Các trạm xăng cung cấp tối đa 4, 2, 5 và 10 đơn vị xăng theo thứ tự.

Cách đi tối ưu là: đi 10 đơn vị, lấy 10 đơn vị xăng, đi tiếp 4 đơn vị, lấy thêm 5 đơn vị xăng, sau đó đi thẳng đến thị trấn.

Sản xuất sữa chua #2 [yoghurt2]

Dioxin Milk là hãng sữa Mĩ nổi tiếng trên toàn thế giới, gần đây họ đầu tư dây chuyền sản xuất sữa chua. Các kĩ sư của hãng cho biết, hiện tại trong kho của nhà máy không còn hộp sữa chua nào và

trong N ngày tới, ngày thứ i hãng sản xuất được a_i hộp. Bộ phận bán hàng cho biết, đơn đặt hàng của N ngày tới tương ứng là b_i hộp cho ngày thứ i .

Vì mới tham gia vào thị trường sữa chua, Dioxin Milk cần thỏa mãn nhiều đơn hàng nhất có thể, khi đơn hàng cho một ngày nào đó bị từ chối, số hộp sữa còn lại của ngày hôm đó (bao gồm số sữa mới sản xuất trong ngày và số sữa còn tồn từ ngày hôm trước) được bảo quản cho ngày hôm sau.

Dựa trên thông tin sản xuất và thông tin về các đơn hàng, hãy xác định số đơn hàng nhiều nhất mà Dioxin Milk có thể đáp ứng và đưa ra một phương án phục vụ.

Dữ liệu (yoghurt2.inp)

- Dòng 1: số nguyên N ($1 \leq N \leq 250000$)
- Dòng 2: N số nguyên a_1, a_2, \dots, a_N ($0 \leq a_i \leq 10^9 \forall i$)
- Dòng 3: N số nguyên b_1, b_2, \dots, b_N ($0 \leq b_i \leq 10^9 \forall i$)

Kết quả (yoghurt2.out)

- Dòng 1: số nguyên K là số đơn hàng đáp ứng được nhiều nhất
- Dòng 2: K số nguyên theo thứ tự tăng là danh sách số thứ tự ngày của các đơn hàng được đáp ứng. Nếu có nhiều phương án thỏa mãn thì chỉ cần đưa ra một phương án bất kì.

Ví dụ

yoghurt2.inp
6
2 2 1 2 1 0
1 2 2 3 4 4

yoghurt2.out
3
1 2 4

Kéo xe [cardrag]

Bãi đỗ xe ô tô bị thu giữ vì vi phạm luật giao thông có dạng lưới ô vuông kích thước $M \times N$, mỗi xe nằm gọn trên một ô vuông theo một trong bốn hướng N, S, E, W (bắc, nam, đông, tây).

Vì một số nguyên nhân đặc biệt, quản lý bãi đỗ cần giải phóng nhanh chóng càng nhiều xe càng tốt. Muốn vậy cần đẩy các xe ra khỏi bãi mà không gây va chạm, tức là cần đẩy các xe theo hướng đi ban đầu đi ra khỏi bãi sao cho trong quá trình đẩy xe không đi vào ô vẫn còn xe. Một xe nếu đã bị di chuyển thì phải được kéo ra khỏi bãi.

Hãy xác định số lượng xe nhiều nhất có thể đẩy ra khỏi bãi với một trình tự đẩy xe tối ưu.

Dữ liệu

Gồm nhiều tests, mỗi test cho trên nhóm dòng theo định dạng:

- Dòng 1: hai số nguyên M, N ($1 \leq M, N \leq 2000$);
- Dòng 2 ... $M + 1$: mỗi dòng ghi xâu độ dài N chỉ gồm các kí tự trong tập $\{ \cdot, N, S, E, W \}$ thể hiện ô trống hay ô có xe đỗ theo hướng tương ứng.

File dữ liệu kết thúc bằng dòng ghi hai số 0.

Kết quả

Kết quả mỗi test ghi trên một dòng số nguyên là số lượng xe nhiều nhất có thể đẩy ra khỏi bãi.

Ví dụ

cardrag.inp	cardrag.out
3 4 .N.W WWSS EWEW 0 0	4

Cân voi (scales)

Chán cảnh đi chăn trâu, Cuội bỏ lên miền núi chăn voi thuê cho nhà thống lí một bản nọ. Thống lí có một cô con gái rất xinh đẹp, lão "câu như" Cuội bằng cách giao hẹn nếu chăn đàn voi tốt thì sẽ gả con gái cho. Trờì chiều lòng người, đàn voi Cuội chăn lớn nhanh như thổi. Giờ đây Cuội chỉ còn cân lũ voi nữa là ... xong.

Cuội được sử dụng một cân thăng bằng và N quả nặng đã biết khối lượng. Cuội cân voi bằng cách đưa voi lên một bàn cân và chất các quả nặng lên bàn cân bên kia cho đến khi thăng bằng. (Cuội không đưa quả nặng lên bàn cân có voi được vì lũ voi này rất hiếu chiến, chúng sẽ dùng vòi hất ngay các quả nặng đi). Chiếc cân cũng có giới hạn, nó sẽ gãy nếu Cuội đặt lên một trong hai bàn cân tổng khối lượng vượt quá C .

Các quả nặng có một tính chất khá thú vị là nếu sắp chúng thành một hàng theo khối lượng tăng dần thì mỗi quả (từ thứ ba trở đi) sẽ không nhẹ hơn tổng khối lượng hai quả liền trước. Để tránh khả năng bị thống lí "đánh tháo" vì làm gãy cân, Cuội muốn biết khối lượng lớn nhất có thể cân được là bao nhiêu. Hãy lập trình tính toán giúp Cuội.

Dữ liệu (scales.inp)

- Dòng 1: hai số nguyên N C ($1 \leq N \leq 1000$; $1 \leq C < 2^{30}$).
- Dòng 2 ... $N + 1$: mỗi dòng một số nguyên trong phạm vi số nguyên 31 bit có dấu là khối lượng của một quả nặng, theo thứ tự không giảm.

Kết quả (scales.out)

- Dòng 1: số nguyên chỉ khối lượng lớn nhất có thể cân được chính xác và an toàn.

Ví dụ

scales.inp	scales.out
3 15 1 10 20	11

Tàu điện (inftrain)

Tuyến tàu điện ZZZ có các ga được đánh số 0,1,2, ... theo thứ tự hành trình. Có tất cả N tàu điện hoạt động trên tuyến này. Tàu điện i có giới hạn số lượng hành khách c_i , xuất phát từ ga 0 chạy đến ga s_i . Các tàu điện đều xuất phát cùng một thời điểm và có thể đón, trả khách tại tất cả các ga mà nó đi qua.

Có M hành khách đang đợi tàu, hành khách i muốn lên tàu ở ga a_i và xuống ở ga b_i , mỗi hành khách chỉ đi một tàu.

Hãy xác định một phương án đón trả khách (hành khách nào lên tàu nào) sao cho đáp ứng được nhu cầu di chuyển của nhiều hành khách nhất.

Dữ liệu

- Dòng 1: hai số nguyên N, M ($1 \leq N, M \leq 10^5$)
- Dòng 2 ... $N + 1$: dòng $i + 1$ ghi hai số nguyên s_i, c_i ($1 \leq s_i, c_i \leq 10^9 \forall i$)
- Dòng $N + 2$... $N + M + 1$: dòng $i + N + 1$ ghi hai số nguyên a_i, b_i ($1 \leq a_i \leq b_i \leq 10^9 \forall i$).

Kết quả

- Dòng 1: số nguyên là số lượng hành khách phục vụ được.
- Dòng 2 ... $N + 1$: dòng $i + 1$ ghi số nguyên là số hiệu tàu mà hành khách i lên, số này bằng 0 nếu hành khách này không được phục vụ.

Ví dụ

inftrain.inp	inftrain.out
2 3	3
10 1	2
15 1	1
2 8	2
7 10	
8 13	

Chấm điểm

- Nếu chỉ đưa ra đúng số lượng hành khách, bạn được 30% điểm của test;
- 20% điểm của bài dành cho các test có $N = 1$.

Thay xâu [strwc]

Bờm có xâu S độ dài N và số nguyên L , xâu S chỉ gồm các chữ cái $a \dots z$ và kí tự $*$. Bờm muốn thay mỗi kí tự $*$ bằng một chữ cái nào đó sao cho trong xâu thu được có xâu con là ghép liên tiếp của một xâu độ dài L .

Hãy giúp Bờm tính toán độ dài xâu con dài nhất có thể tạo được

Dữ liệu (strwc.inp)

- Dòng 1: số nguyên T ($1 \leq T \leq 10$) là số test
- Mỗi test cho trên hai dòng, dòng thứ nhất chứa hai số N, L ($1 \leq N \leq 100000; 1 \leq L \leq N$), dòng thứ hai ghi xâu S .

Kết quả (strwc.out)

- Dòng 1 ... T : dòng i ghi số nguyên kết quả của test thứ i

Ví dụ

strwc.inp	strwc.out
1	6
7 3	
a*cab*d	

Trình diễn xe hơi #1 (carshow1)

Hãng ZZZ tổ chức một buổi trình diễn N mẫu xe hơi mới của hãng. N xe sẽ cùng chạy trên một tuyến đường coi như trục tọa độ, xe i xuất phát ở thời điểm 0 từ vị trí x_i với vận tốc v_i đơn vị dài/giây chạy theo hướng dương của trục tọa độ, vị trí xuất phát của các xe là đôi một phân biệt.

Buổi trình diễn kéo dài trong T giây, vì đường chỉ có một làn nên khi có xe i đuổi kịp xe j , xe i cần giảm tốc độ bằng với xe j và chúng sẽ nối đuôi nhau chạy tiếp, coi kích thước các xe là không đáng kể. Như vậy ở mỗi thời điểm, trên đường sẽ có các nhóm xe chạy nối đuôi nhau mà ở thời điểm 0 có N nhóm (mỗi nhóm 1 xe).

Hãy xác định số nhóm xe chạy nối đuôi nhau ở thời điểm T .

Dữ liệu

- Dòng 1: hai số nguyên N, T ($1 \leq N \leq 10^5$; $1 \leq T \leq 10^9$)
- Dòng 2 ... $N + 1$: dòng $i + 1$ ghi hai số nguyên x_i, v_i ($0 \leq x_i$; $0 < v_i$; $x_i, v_i \leq 10^9 \forall i$), các giá trị x_i là đôi một phân biệt, các xe được cho theo thứ tự x_i tăng.

Kết quả

- Dòng 1: số nguyên kết quả.

Ví dụ

carshow1.inp	carshow1.out
5 3 0 1 1 2 2 3 3 2 6 1	3

Xây dựng nhà máy (build)

Dọc theo một tuyến đường cao tốc mà ta coi như một trục tọa độ, người ta dự định xây dựng N nhà máy đánh số $1 \dots N$, sao cho với $i < j$ thì tọa độ xây dựng nhà máy i phải không nhỏ hơn tọa độ xây dựng nhà máy j .

Do cách thức sản xuất, các mối quan hệ về nguyên liệu, nhiên liệu hay môi trường mà có một số cặp nhà máy không được xây dựng quá xa nhau, cũng có một số cặp nhà máy không được quá gần nhau. Cụ thể có M điều kiện về khoảng cách lớn nhất có thể giữa các cặp nhà máy; P điều kiện về khoảng cách nhỏ nhất có thể giữa các cặp nhà máy.

Hãy xác định khoảng cách lớn nhất có thể giữa nhà máy 1 và nhà máy N .

Dữ liệu (build.inp)

- Dòng 1: ba số nguyên $N M P$ ($2 \leq N \leq 1,000$; $1 \leq M \leq 10,000$; $1 \leq P \leq 10,000$).
- Dòng 2 ... $M + 1$: mỗi dòng ba số nguyên $A B D$ ($1 \leq A < B \leq N$) là điều kiện hai nhà máy A, B không được xa nhau quá D ($1 \leq D \leq 1,000,000$).
- Dòng $M + 2 \dots M + P + 1$: mỗi dòng ba số nguyên $A B D$ ($1 \leq A < B \leq N$) là điều kiện hai nhà máy A, B không được xa nhau ít hơn D ($1 \leq D \leq 1,000,000$).

Kết quả (build.out)

- Dòng 1: một số nguyên, nếu không có cách xây dựng ghi -1 , nếu nhà máy 1 và nhà máy N có thể cách xa tùy ý ghi -2 , ngược lại ghi khoảng cách lớn nhất tìm được

Ví dụ

build.inp	build.out
4 2 1 1 3 10 2 4 20 2 3 3	27

Trọng số âm [nonneg]

Cho đồ thị gồm N đỉnh, M cạnh vô hướng và P cung có hướng, các đỉnh đánh số $1, 2, \dots, N$. Trọng số các cạnh không âm, còn trọng số các cung có thể âm. Ngoài ra, đồ thị này có tính chất đặc biệt là nếu có cung từ đỉnh u đến đỉnh v thì sẽ không có đường đi từ v đến u .

Hãy xác định độ dài đường đi ngắn nhất từ một đỉnh S cho trước đến từng đỉnh của đồ thị.

Dữ liệu (nonneg.inp)

- Dòng 1: bốn số nguyên N, M, P, S ($1 \leq N \leq 25000; 1 \leq M, P \leq 50000$)
- Dòng 2 ... $M + 1$: mỗi dòng ba số nguyên u, v, c ($0 \leq c \leq 10000$) thể hiện cạnh nối hai đỉnh u, v có trọng số c .
- Dòng $M + 2$... $M + P + 1$: mỗi dòng ba số nguyên u, v, c ($0 \leq |c| \leq 10000$) thể hiện cung nối từ đỉnh u đến đỉnh v có trọng số c .

Kết quả (nonneg.out)

- Dòng 1 ... N : dòng i ghi trọng số đường đi ngắn nhất từ S đến i , nếu không có đường đi thì ghi xâu "NO PATH"

Ví dụ

nonneg.inp	nonneg.out
6 3 3 4 1 2 5 3 4 5 5 6 10 3 5 -100 4 6 -100 1 3 -10	NO PATH NO PATH 5 0 -95 -100

Mê cung thời gian (wormhole)

Để giúp Winky khuây khỏa nỗi buồn sau khi bị ông Crouch đuổi, Dobby quyết định đưa cô nàng gia tinh đến chơi ở mê cung thời gian trong trường Hogwarts. Mê cung này gồm N căn phòng nối với nhau bởi M hành lang và W cầu thang: hành lang thứ i là hai chiều nối căn phòng A_i với căn phòng B_i , thời gian di chuyển là T_i ; cầu thang thứ j là một chiều đưa Winky từ căn phòng C_j đến căn phòng D_j vào thời điểm trước khi bước vào cầu thang một khoảng thời gian U_j .

Winky nhanh chóng khoái mê cung này và quyết định sẽ xuất phát từ một căn phòng đi qua một số hành lang và cầu thang rồi lại trở về căn phòng ban đầu vào thời điểm trước khi xuất phát (Như thế rất có thể trong lộ trình Winky sẽ gặp chính mình)

Dobby đã kiến thiết F mê cung nhưng cậu chàng không biết mỗi mê cung có thỏa yêu cầu của Winky hay không, hãy giúp chàng gia tình lấy lòng Winky bằng cách đưa ra câu trả lời cho Dobby.

Dữ liệu (wormhole.inp)

- Dòng 1: số nguyên F ($1 \leq F \leq 5$), tiếp theo là mô tả của mỗi mê cung:
- Dòng 1 của mỗi mê cung: ba số nguyên N, M, W ($1 \leq N \leq 500$; $1 \leq M \leq 2,500$; $1 \leq W \leq 200$)
- Dòng 2 ... $M + 1$ của mỗi mê cung: dòng $i + 1$ gồm 3 số A_i, B_i, T_i ($0 \leq T_i \leq 10,000$)
- Dòng $M + 2$... $M + W + 1$ của mỗi mê cung: dòng $j + M + 1$ gồm 3 số C_j, D_j, U_j ($0 \leq U_j \leq 10,000$)

Kết quả (wormhole.out)

- Dòng 1 ... F : với mỗi mê cung ghi trên một dòng câu trả lời "YES" hay "NO".

Ví dụ

wormhole.inp	wormhole.out
2 3 3 1 1 2 2 1 3 4 2 3 1 3 1 3 3 2 1 1 2 3 2 3 4 3 1 8	NO YES

Quà khuyến mại (promotion)

Hội chợ nọ có N gian hàng đánh số thứ tự $1, 2, \dots, N$. Gian hàng i có phát quà khuyến mại trị giá f_i , quà chỉ được phát cho mỗi khách hàng một lần ở lần thăm đầu tiên. Có M đường đi một chiều nối trực tiếp các cặp gian hàng, đường đi thứ j nối hai gian hàng u_j, v_j có thời gian di chuyển t_j .

Chàng lười Z muốn xuất phát từ một gian hàng, đi thăm hội chợ thêm ít nhất một gian hàng nữa để nhận quà khuyến mại rồi quay về gian hàng xuất phát. Vì lười nên anh ta cần tìm một hành trình cho tỉ số giữa tổng giá trị quà thu được và thời gian di chuyển là lớn nhất.

Hãy xác định tỉ số lớn nhất đó.

Dữ liệu (promotion.inp)

- Dòng 1: hai số nguyên N, M ($2 \leq N \leq 1000$; $2 \leq M \leq 5000$)
- Dòng 2: N số nguyên f_1, f_2, \dots, f_N ($1 \leq f_i \leq 1000 \forall i$)
- Dòng 3 ... $M + 2$: dòng $i + 2$ ghi ba số nguyên a_i, b_i, t_i ($1 \leq t_i \leq 1000$)

Kết quả (promotion.out)

- Dòng 1: số thực với hai chữ số ở phần thập phân là tỉ số lớn nhất thu được, nếu không thể có hành trình thỏa mãn yêu cầu thì dòng này chỉ ghi số 0.

Ví dụ

promotion.inp	promotion.out
---------------	---------------

5 7 30 10 10 5 10 1 2 3 2 3 2 3 4 5 3 5 2 4 5 5 5 1 3 5 2 2	6.00
---	------

Giải thích

Hành trình $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 1$.

Thay xâu [strwc]

Bờm có xâu S độ dài N và số nguyên L , xâu S chỉ gồm các chữ cái $a \dots z$ và kí tự $*$. Bờm muốn thay mỗi kí tự $*$ bằng một chữ cái nào đó sao cho trong xâu thu được có xâu con là ghép liên tiếp của một xâu độ dài L .

Hãy giúp Bờm tính toán độ dài xâu con dài nhất có thể tạo được

Dữ liệu (strwc.inp)

- Dòng 1: số nguyên T ($1 \leq T \leq 10$) là số test
- Mỗi test cho trên hai dòng, dòng thứ nhất chứa hai số N, L ($1 \leq N \leq 100000; 1 \leq L \leq N$), dòng thứ hai ghi xâu S .

Kết quả (strwc.out)

- Dòng 1 ... T : dòng i ghi số nguyên kết quả của test thứ i

Ví dụ

strwc.inp	strwc.out
1 7 3 a*cab*d	6

Biểu diễn số [partkx]

Cho số nguyên dương N , tìm số cách phân tích N thành tổng K số nguyên dương mà hai số bất kì trong đó chênh lệch không ít hơn X . Hai cách phân tích được coi là khác nhau nếu trong cách này có số hạng không xuất hiện trong cách kia.

Dữ liệu (partkx.inp)

- Dòng 1: ba số nguyên N, K, X ($0 < N \leq 10000; 1 \leq K, X \leq N$).

Dữ liệu (partkx.out)

- Dòng 1: số nguyên là phần dư của số cách phân tích khi chia cho 10007

Ví dụ

partkx.inp	partkx.out
20 2 3	8

Găng tay [sgloves]

Quốc vương nước zZz rất thích găng tay, ông có hai gian phòng để cất giữ găng tay, một để găng tay trái, phòng kia để găng tay phải, mỗi chiếc đều được để trong một hộp gỗ, các hộp đều giống nhau.

Mỗi găng tay của quốc vương có màu là một trong N màu đánh số $1, 2, \dots, N$; số găng tay trái mỗi màu lần lượt là a_1, a_2, \dots, a_N ; số găng tay phải mỗi màu lần lượt là b_1, b_2, \dots, b_N . Vấn đề là các hộp chứa găng đều không có kí hiệu cho biết trong đó là găng màu gì. Vì vậy, mỗi khi quốc vương ra ngoài, người hầu của quốc vương phải vào mỗi phòng lấy ra một số hộp để quốc vương mở ra và chọn lấy cặp găng cùng màu.

Hãy giúp người hầu của quốc vương xác định một cách lấy hộp ở hai phòng sao cho quốc vương chắc chắn chọn được đôi găng tay cùng màu và tổng số hộp cần lấy là nhỏ nhất.

Dữ liệu (sgloves.inp)

- Dòng 1: số nguyên N ($1 \leq N \leq 20$)
- Dòng 2: N số nguyên a_1, a_2, \dots, a_N ($0 \leq a_i \leq 10^8$)
- Dòng 3: N số nguyên b_1, b_2, \dots, b_N ($0 \leq b_i \leq 10^8$), dữ liệu đảm bảo tồn tại j sao cho $a_j > 0$ và $b_j > 0$

Kết quả (sgloves.out)

- Dòng 1: hai số nguyên l, r là số hộp cần lấy ở hai phòng chứa găng.

Ví dụ

sgloves.inp	sgloves.out
4	2 8
0 7 1 6	
1 5 0 6	

Đường đi nhỏ nhất [mlexpath]

Cho bảng số nguyên kích thước $M \times N$, các hàng đánh số $1 \dots M$, các cột đánh số $1 \dots N$. Các số trong bảng là đôi một phân biệt và đều thuộc phạm vi $1 \dots M \times N$. Điều kiện di chuyển trong lưới là: từ một ô chỉ được đi sang ô kề phải hoặc ô kề dưới. Cần tìm đường đi từ ô trái trên $(1; 1)$ đến ô phải dưới $(M; N)$ sao cho dãy số trên đường đi sau khi sắp xếp tăng có thứ tự từ điển nhỏ nhất có thể.

Dữ liệu

- Dòng 1: hai số nguyên M, N ($1 \leq M, N \leq 1,500$)
- Dòng 2 ... $M + 1$: mỗi dòng ghi N số nguyên, số thứ j trên dòng $i + 1$ là số trong ô $(i; j)$ của lưới.

Kết quả

- Dòng 1: dãy số nguyên lần lượt là số trong các ô trên đường đi tìm được, tính cả ô xuất phát và ô kết thúc.

Ví dụ

mlexpath.inp	mlexpath.out
4 4 7 4 13 3 8 11 12 2 10 9 1 5 16 14 15 6	7 4 11 9 1 5 6