

1. STRING

Một xâu kí tự X có thể chuyển đổi được sang xâu kí tự Y cùng độ dài nếu 2 xâu chỉ khác nhau đúng 1 vị trí duy nhất. Ví dụ: $X = abcd$ và $Y = abdc$ không thể chuyển đổi được cho nhau vì chúng khác nhau ở kí tự thứ 3 và 4. $X = abcd$ và $Y = abce$ có thể chuyển đổi được cho nhau vì chúng chỉ khác nhau ở kí tự thứ 4.

Yêu cầu: Bạn được cho 2 xâu kí tự S, T và N xâu kí tự trong từ điển có cùng độ dài K . Nhiệm vụ của bạn là tìm cách biến đổi từ xâu S sang xâu T dựa vào N xâu trong từ điển (tức là không được dùng bất kỳ xâu kí tự nào ngoài N xâu trong từ điển để thực hiện phép chuyển đổi) sao cho số lượng phép chuyển đổi là ít nhất có thể.

Dữ liệu: Vào từ file STRING.INP gồm

- Dòng đầu tiên chứa 2 số nguyên N, K ($1 \leq N \leq 2000, 1 \leq K \leq 20$).
- Dòng tiếp theo chứa xâu S độ dài K .
- Dòng tiếp theo chứa xâu T độ dài K .
- N dòng tiếp theo chứa N xâu độ dài K trong từ điển.
- (Tất cả các xâu chỉ chứa các ký tự latin từ 'a' đến 'z')

Kết quả: Ghi ra file STRING.OUT 2 số nguyên d và t , trong đó d là số lượng phép chuyển đổi ít nhất phải thực hiện, t là số lượng các cách biến đổi khác nhau với d phép chuyển đổi để chuyển đổi từ xâu S sang xâu T . Vì số lượng cách chuyển đổi có thể rất lớn nên bạn chỉ cần in ra số lượng cách chuyển đổi module $10^9 + 7$. (Nếu không có đáp án thì ghi ra hai số -1 -1)

Ví dụ:

STRING.INP	STRING.OUT	Giải thích
2 3 abc add abd adc	2 2	Phải thực hiện ít nhất 2 phép chuyển đổi để chuyển đổi từ $S=abc$ sang $T=add$, và có 2 cách biến đổi dùng ít nhất 2 phép chuyển đổi: $abc \rightarrow abd \rightarrow add$ hoặc $abc \rightarrow adc \rightarrow add$.

Giới hạn: Trong tất cả các test thì có 30% số test với $N \leq 10$; 30% số test với $N \leq 200$.

2. HỘI CHỢ

Khu hội chợ Đông Bắc Bắc Giang có $m \times n$ gian hàng được bố trí trong một khu hình chữ nhật kích thước $m \times n$. Các hàng của hình chữ nhật được đánh số $1, 2, 3, \dots, m$ từ trên xuống dưới, còn các cột – đánh số $1, 2, 3, \dots, n$ từ trái sang phải, ô nằm giao của hàng i và cột j là gian hàng (i, j) trưng bày mặt hàng a_{ij} . Khách tham quan đi vào khu hội chợ từ một gian hàng bất kỳ bên trái (i bất kỳ, $j = 1$) và mất 1 đồng, không nhất thiết phải tham quan tất cả các gian hàng, khách chỉ có thể đi ra khỏi khu hội chợ từ các gian hàng bên phải

(i bất kỳ, $j = n$), tại mỗi gian hàng khách có thể di chuyển qua các gian hàng chung cạnh với nó. Khi đi vào gian hàng trưng bày mặt hàng khác với mặt hàng của gian hàng hiện tại thì khách tham quan phải mua vé giá là 1 đồng.

Yêu cầu: Cho biết mặt hàng trưng bày tại các gian hàng, tính chi phí ít nhất mà khách tham quan phải trả khi tham quan khu hội chợ.

Dữ liệu: Vào từ file văn bản FAIR.INP:

- Dòng đầu tiên ghi hai số m, n ;
- m dòng sau, mỗi dòng n số nguyên không âm, cho mã mặt hàng được trưng bày tại các gian hàng của khu hội chợ. Mã mặt hàng tại gian hàng (i, j) là a_{ij} thỏa mãn $0 \leq a_{ij} \leq 100$.

Hai số liên tiếp trên một dòng cách nhau một dấu cách.

Kết quả: Ghi ra file văn bản FAIR.OUT gồm một số duy nhất là chi phí ít nhất tìm được.

Ví dụ:

FAIR.INP	FAIR.OUT
2 3 0 1 1 1 1 2	1

Ràng buộc:

- Có 30% số test ứng với 30% số điểm của bài có $m, n \leq 5$;
- Có 30% số test ứng với 30% số điểm của bài có $m, n \leq 50$;
- Có 40% số test khác ứng với 40% số điểm còn lại của bài có $m, n \leq 1000$.

3. ROBOT

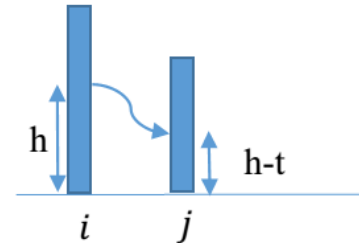
HD vừa sáng tạo ra một trò chơi điều khiển robot mới cho 2 bé Bi, Bo chơi. Nội dung trò chơi như sau:

- Có N cây cột đánh số từ 1 đến N , cây cột thứ i có chiều cao $h[i](m)$
- Có M đường nhảy dạng i, j, t tương ứng là nhảy từ cây i sang cây j (hoặc từ cây j sang cây i) mất $t(s)$ và nếu nhảy từ độ cao h ($h \in \mathbb{N}, h \leq h[i]$) của cây i thì sang cây j sẽ có độ cao là $h - t$ với điều kiện $0 \leq h - t \leq h[j]$
- Nếu robot di chuyển lên xuống trên cột hiện tại, thời gian di chuyển mất $1(s)$ trên $1m$ di chuyển.

Hiện tại robot đang ở độ cao X của cây 1, Bi-Bo cần phải tìm phương án di chuyển nhanh nhất đến độ cao $h[N]$ của cây N . Bạn hãy giúp 2 bé Bi-Bo tính thời gian di chuyển ngắn nhất thỏa mãn yêu cầu đầu bài?

Dữ liệu: Vào từ file văn bản ROBOT.INP

- Dòng 1: Chứa 3 số nguyên dương N, M, X tương ứng là số lượng cây cột, số lượng đường nhảy và độ cao của robot đang ở cột 1. ($2 \leq N \leq 100.000$; $1 \leq M \leq 300.000$; $0 \leq X \leq h[1]$)
- N dòng tiếp theo, dòng thứ i chứa 1 số nguyên dương $h[i]$ tương ứng là chiều cao của cột i ($1 \leq h[i] \leq 1.000.000.000$ $\forall i = 1..N$).
- M dòng tiếp theo, mỗi dòng chứa 3 số nguyên dương i, j, t tương ứng là nhảy từ cây i sang cây j (hoặc từ cây j sang cây i) mất $t(s)$ ($1 \leq t \leq 1.000.000.000$).



Các số trên một dòng của input file được ghi cách nhau bởi dấu cách.

Kết quả: Ghi ra file văn bản ROBOT.OUT một số duy nhất là thời gian ngắn nhất để robot di chuyển đến độ cao $h[N]$ của cây N , nếu không thể di chuyển đến thì ghi -1.

Ví dụ:

ROBOT.INP	ROBOT.OUT	Giải thích
5 5 0 50 100 25 30 10 1 2 10 2 5 50 2 4 20 4 3 1 5 4 20	110	Trèo lên 50(m) ở cây 1 mất 50(s) Nhảy từ cây 1 sang cây 2: - Mất 10(s) - ở độ cao 40 trên cây 2 Nhảy từ cây 2 sang cây 4: - Mất 20(s) - ở độ cao 20 trên cây 4 Nhảy từ cây 4 sang cây 5: - Mất 20(s) - ở độ cao 0 - trèo thêm 10(m) mất 10(s) Tổng thời gian: 110(s).
2 1 0 1 1 1 2 100	-1	Từ cây 1, bất kỳ độ cao nào, khi nhảy sang cây 2 đều không thực hiện được vì $h - t < 0$.
4 3 30 50 10 20 50 1 2 10 2 3 10 3 4 10	100	Di chuyển xuống 10(m) ở cây 1 mất 10 (s) và đang ở độ cao 20(m) Nhảy sang cây 2: - Mất 10(s) - ở độ cao 10 trên cây 2. Nhảy sang cây 3: - Mất 10(s) - Ở độ cao 0(m), trèo lên 10(m) mất 10(s),

		<p>ở độ cao 10(m);</p> <p>Nhảy sang cây 4:</p> <ul style="list-style-type: none"> - Mất 10(s), - ở độ cao 0 (m), trèo lên 50(m) mất 50(s) <p>Tổng thời gian: 100(s).</p>
--	--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------