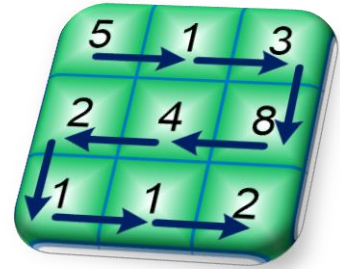


TRƯỢT PATANH

Tên chương trình: ROLLER.???

Steve cùng bạn bè đi đến công viên giải trí để thư giãn. Steve rất ghét trượt pa tanh vì vậy chỉ ngồi xem các bạn trượt.

Sân trượt có hình chữ nhật, chia thành các ô vuông gồm r hàng và c cột ($2 \leq r, c \leq 10^3$). Mỗi ô vuông có ghi một số nguyên dương có giá trị không quá 1 000. Người trượt xuất phát từ ô trên trái của sân và đi tới đích là ô dưới phải, từ một ô có thể trượt sang ô kề cạnh, mỗi ô được đi qua không quá một lần. Điểm số của lần trượt là tổng các số ghi trên những ô đã đi qua.



Quan sát sân trượt, Steve ngồi nhằm tính đường đi để đạt được số điểm cao nhất.

Yêu cầu: Cho r , c và số ghi trên các ô. Bằng các ký tự từ tập $\{U, D, L, R\}$ hãy chỉ ra một đường đi có điểm số cao nhất.

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

- Dòng đầu tiên chứa 2 số nguyên r và c ,
- Mỗi dòng trong r dòng sau chứa c số nguyên xác định các số ghi trên một dòng từ trái sang phải.

Kết quả: Đưa ra file văn bản ROLLER.OUT xâu xác định đường đi có điểm cao nhất.

Ví dụ:

ROLLER.INP	
3	3
5	1 3
2	4 8
1	1 2

ROLLER.OUT
RRDLLDRR

SẮP XẾP SÁCH

Tên chương trình: BOOKSORT.???

Steve có một thư viện nhỏ gồm n cuốn sách. Do phòng hẹp, sách phải một chồng cao. Sách được đánh số từ 1 đến n theo thứ tự từ điển của tên sách. Ban đầu, các cuốn sách nằm đúng vị trí của mình từ trên xuống dưới theo đúng trình tự từ 1 đến n .

Khi cần một cuốn sách nào đó, Steve dễ dàng rút ra, nhưng sau đó không thể nhét lại vào vị trí cũ mà phải đặt nó lên trên cùng. Thời gian trôi qua và các sách không còn nằm đúng vị trí ban đầu làm cho việc tìm cuốn sách cần thiết trở nên khó khăn. Steve muốn khôi phục lại vị trí ban đầu của các cuốn sách. Để làm việc đó Steve rút ra một cuốn sách, đặt nó lên trên cùng, rồi lại rút ra cuốn khác, đặt lên trên . . . cứ như thế cho đến khi sách được sắp xếp theo thứ tự từ điển từ trên xuống dưới. Ví dụ, với $n = 4$ và trật tự các cuốn sách từ trên xuống dưới là (1, 3, 4, 2), Steve cần rút cuốn 2 đặt lên trên, sau đó rút tiếp cuốn 1 đặt lên trên, tổng cộng là 2 lần rút sách ra đặt lại.



Yêu cầu: Cho n và trình tự hiện tại từ trên xuống dưới của các cuốn sách. Hãy xác định số lần tối thiểu cần rút sách ra để sắp xếp lại sách theo đúng trình tự.

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

- Dòng đầu tiên chứa số nguyên n ($n \leq 300\,000$),
- Dòng thứ i trong n dòng sau chứa một số nguyên xác định số thứ tự của cuốn sách ở vị trí i tính từ trên xuống.

Kết quả: Đưa ra file văn bản BOOKSORT.OUT một số nguyên – số lần tối thiểu rút sách cần thực hiện.

Ví dụ:

BOOKSORT.INP	BOOKSORT.OUT
4	2
1	
3	
4	
2	

TRÒ CHƠI

Tên chương trình: GAME.???

Rôn và Hermione chơi trò chơi xếp chữ. Từ một xâu nguồn độ dài n chỉ chứa các ký tự latin thường hai người lần lượt đi. Đến lượt ai người đó chọn một ký tự hiện có trong xâu ghi vào cuối xâu riêng của mình và xóa ký tự được chọn trong xâu nguồn. Trò chơi kết thúc khi xâu nguồn rỗng. Ban đầu xâu riêng của 2 người đều rỗng. Từ của ai có thứ tự từ điển nhỏ hơn là người ấy thắng. Nếu hai xâu riêng giống nhau thì coi là cả hai cùng thua. Rôn là người đi trước.

Rôn chơi tốt hơn và thường thắng, vì vậy muốn bạn mình không chán Rôn quyết định luôn luôn chỉ chọn ký tự cuối của xâu nguồn. Hermione nhanh chóng nhận ra chiến lược chơi của Rôn và cố gắng tìm cách giành chiến thắng.

Yêu cầu: Cho n ($2 \leq n \leq 10^5$) và xâu nguồn. Hãy xác định xem Hermione có thắng được hay không và đưa ra câu trả lời **YES** hoặc **NO** tương ứng. Trong mọi trường hợp – đưa ra từ riêng mà Hermione nhận được khi chơi theo chiến lược tối ưu.

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

- Dòng đầu tiên chứa số nguyên n ,
- Dòng thứ 2 chứa xâu độ dài n chỉ chứa các ký tự la tinh thường.

Kết quả: Đưa ra file văn bản GAME.OUT:

- Dòng đầu tiên chứa thông báo **YES** hoặc **NO**,
- Dòng thứ 2 chứa xâu riêng của Hermione.

Ví dụ:

GAME.INP
8
cokolada

GAME.OUT
YES
acko