

Tháp Hà nội 1f

Trò chơi tháp Hà nội 1f là trò chơi tổng quát của tháp Hà nội cổ điển gồm n đĩa với n kích thước khác nhau và ba cọc A, B, C. Một ma trận G thể hiện mối quan hệ được phép chuyển đĩa giữa các cọc (cọc x sang cọc y nếu cọc x sang cọc y nếu $G[x][y] = 1$). Trò chơi bắt đầu bằng trạng thái các đĩa được chồng lên nhau ở cọc A.

Yêu cầu của trò chơi là chuyển toàn số đĩa từ cọc A sang cọc C, tuân theo các quy tắc sau:

- Chỉ sử dụng 3 cọc để chuyển;
- Một lần chỉ được di chuyển một đĩa nằm trên cùng từ cọc x sang cọc y nếu cọc x sang cọc y nếu $G[x][y] = 1$;
- Một đĩa chỉ được đặt lên một đĩa không nhỏ hơn.

Yêu cầu: Hãy tìm cách chuyển toàn bộ đĩa thành một chồng đĩa ở cọc C.

Input

- Dòng đầu chứa số nguyên dương n ($n \leq 15$);
- Tiếp theo là ba dòng, mỗi dòng ba số mô tả ma trận G .

Output

- Dòng đầu chứa số nguyên s là số lần chuyển đĩa;
- Dòng thứ j ($j = 1, 2, \dots, s$) trong s dòng tiếp theo, mỗi dòng gồm đúng hai kí tự mô tả một thao tác chuyển đĩa. Cụ thể, kí tự thứ nhất là tên cọc chứa đĩa cần chuyển, kí tự thứ hai là tên cọc mà đĩa chuyển tới.

Input	Output
1	2
0 1 0	AB
0 0 1	BC
1 0 0	

Tháp Hà Nội

Có 3 cọc A, B, C và có N chiếc đĩa đánh số từ 1 đến N có kích thước tương ứng là 1, 2, ..., N. Trạng thái ban đầu cả N chiếc đĩa đều ở cọc A, và đĩa to luôn ở dưới đĩa nhỏ. Xét phương pháp chuyển đĩa sau để chuyển đĩa từ cọc A sang cọc C sao cho *các đĩa to luôn luôn ở dưới các đĩa nhỏ* :

gọi thủ tục Chuyen(N, A, B, C).

Trong đó :

```
void Move(n , c1, c2, c3){
    if (n == 1){
        chuyển đĩa nằm trên cùng c1 sang c3;
    }
    else {
        Move(n-1, c1, c3, c2);
        Move( 1, c1, c2, c3);
        Move(n-1, c2, c1, c3);
    }
}
```

có nghĩa là ta chuyển n đĩa trên cùng của c1 sang c3 thông qua c2.

Vậy ta có 2 bài toán sau để giải quyết:

1. Cho trước 1 số P, $P < 2^N$. Hỏi sau lần gọi thứ P thì trạng thái của N đĩa như thế nào?
2. Cho trước một trạng thái của N đĩa, bạn hãy xét xem trạng thái đó có xuất hiện trong quá trình chuyển đĩa từ A sang C theo phương pháp trên hay không? Nếu có xuất hiện thì đó là sau lần gọi hàm Move thứ bao nhiêu? (Giả sử là số Q)

Input

- + Dòng 1: ghi số N ($N < 101$).
- + Dòng 2: ghi số P.
- + Dòng 3: ghi 1 xâu gồm N ký tự chỉ gồm A,B,C là trạng thái của các đĩa (tất nhiên các đĩa to luôn luôn ở dưới các đĩa nhỏ).

Output

- + Dòng 1: ghi 1 xâu gồm N ký tự là trạng thái đĩa sau lần gọi thứ P.
- + Dòng 2: là số Q. (Q = -1 nếu trạng thái đó không xuất hiện).

Dữ liệu vào	Kết quả ra
3	CBA
2	7
CCC	

Xóa số

Xét dãy vô hạn các số tự nhiên liên tiếp bắt đầu từ 1: 1, 2, 3,... và n số nguyên dương a_1, a_2, \dots, a_n . Trên dãy vô hạn các số tự nhiên này, tiến hành xóa hết các số chia hết cho a_1 , sau đó xóa hết các số chia hết cho a_2 mà chưa được xóa,..., cuối cùng xóa hết các số chia hết cho a_n mà chưa được xóa. Đánh số các số chưa được xóa bắt đầu từ 1, người ta muốn biết số được đánh số thứ k là số nào?

Yêu cầu: Cho dãy số a_1, a_2, \dots, a_n và k , hãy tìm số tự nhiên được đánh số thứ k trên dãy sau khi xóa.

Input

- Dòng đầu chứa số nguyên T là số bộ dữ liệu;
- T nhóm dòng sau, mỗi nhóm có dạng:
 - o Dòng đầu của nhóm chứa hai số nguyên dương n và k ($1 \leq k \leq 10^{15}$);
 - o Dòng thứ hai của nhóm chứa n số nguyên dương a_1, a_2, \dots, a_n ($1 < a_i \leq 10^{15}$).

Output

- Gồm T dòng, mỗi dòng chứa một số tự nhiên là kết quả tương ứng của bộ test trong dữ liệu vào.

Input	Output
2	9
1 5	5
2	
2 2	
2 3	

Subtask 1: $n=1$;

Subtask 2: $n=2$;

Subtask 3: $n \leq 10$.

Puzzle of numbers

Xóa một số chữ số trong đẳng thức đúng của tổng hai số nguyên (các số bị xóa được thay bởi các dấu sao “*”), ta có một câu đố như sau: Đếm số lượng cách thay các dấu sao bởi các chữ số để cho đẳng thức vẫn đúng.

Ví dụ, bắt đầu từ đẳng thức sau:

9334

789

10123 (9334+789=10123)

Các ví dụ các chữ số bị mất được thay bằng các dấu sao như sau:

*3*4 hay *****

78* ***

10123 *****

Yêu cầu: Đếm số lượng cách khác nhau thay các dấu sao thành các chữ số để được một đẳng thức đúng, chú ý rằng các số không có chữ số 0 vô nghĩa ở đầu.

Dữ liệu vào trong file “REBUSS.INP”: gồm 3 dòng, mỗi dòng là một xâu kí tự gồm các chữ số hoặc ký tự “*”. Độ dài mỗi xâu không quá 50 kí tự. Dòng 1, dòng 2 thể hiện là hai số hạng, dòng 3 thể hiện là tổng hai số.

Kết quả ra file “REBUSS.OUT”: Gồm một số nguyên là số cách đếm được chia dư cho $(10^9 + 7)$.

Dữ liệu vào	Kết quả ra
*3*4 78* 1012*	10
1* 1* 21	2

Subtask 1: các số có không quá 10 chữ số và mỗi số bị xóa không quá 1 chữ số;

Subtask 2: các số có không quá 50 chữ số và mỗi số bị xóa không quá 3 chữ số;

Subtask 3: các số có không quá 50 chữ số.

Cắt bánh

Một chiếc bánh có dạng là một đa giác đều gồm n đỉnh, các đỉnh được đánh số từ 1 đến n theo chiều kim đồng hồ. Tiến hành m lần cắt bánh, lần cắt thứ k cắt đi qua hai đỉnh phân biệt i_k, j_k .

Yêu cầu: Đếm số miếng bánh sau khi thực hiện m lần cắt bánh.

Dữ liệu: Vào từ thiết bị nhập chuẩn:

- Dòng đầu chứa hai số nguyên n, m ($n \leq 50$);
- Dòng thứ k chứa hai số nguyên i_k, j_k mô tả lần cắt thứ k ($1 \leq i_k, j_k \leq n$ và $i_k \neq j_k$).

Kết quả: Ghi ra thiết bị ra chuẩn một dòng chứa một số nguyên là số miếng bánh sau khi thực hiện m lần cắt bánh.

Ví dụ:

Dữ liệu vào	Kết quả ra
4 1 1 3	2