

TRÁO BÀI

Cho bộ bài gồm n lá bài được xếp thành dãy thứ tự từ 1 tới n , đầu tiên người ta ghi vào mỗi lá bài một số nguyên là số thứ tự ban đầu của lá bài đó. Xét phép tráo $S(i, m, j)$: Lấy ra khỏi bộ bài m lá bài liên tiếp bắt đầu từ lá bài thứ i , sau đó chèn m lá bài này vào trước lá bài thứ j trong số $n - m$ lá bài còn lại $1 \leq i, j \leq n - m + 1$. Quy ước rằng nếu $j = n - m + 1$ thì m lá bài lấy ra sẽ được đưa vào cuối dãy.

Ví dụ với $n = 9$:

Bộ bài ban đầu: (1,2,3,4,5,6,7,8,9)

Thực hiện $S(1,5,2)$: (1,2,3,4,5, 6,7,8,9) \rightarrow (6, 1,2,3,4,5, 7,8,9)

Thực hiện tiếp $S(5,4,6)$: (6,1,2,3, 4,5,7,8, 9) \rightarrow (6,1,2,3,9, 4,5,7,8)

Thực hiện tiếp $S(8,2,1)$: (6,1,2,3,9,4,5, 7,8) \rightarrow (7,8, 6,1,2,3,9,4,5)

Yêu cầu: Hãy cho biết số ghi trên k lá bài đầu tiên của bộ bài ($k \leq n$) sau khi thực hiện x phép tráo bài cho trước.

Dữ liệu: Vào từ file văn bản SHUFFLE.INP

- Dòng 1: Chứa ba số nguyên dương n, k, x ($n \leq 10^5, k \leq 60, x \leq 10^5$)
- x dòng tiếp theo, mỗi dòng ghi ba số nguyên i, m, j tương ứng với một phép tráo $S(i, m, j)$

Kết quả: Ghi ra file văn bản SHUFFLE.OUT một dòng chứa k số nguyên, số thứ i là số ghi trên lá bài thứ i sau khi thực hiện x phép tráo đã cho.

Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách.

Ví dụ:

SHUFFLE.INP	SHUFFLE.OUT
9 2 3	7 8
1 5 2	
5 4 6	
8 2 1	

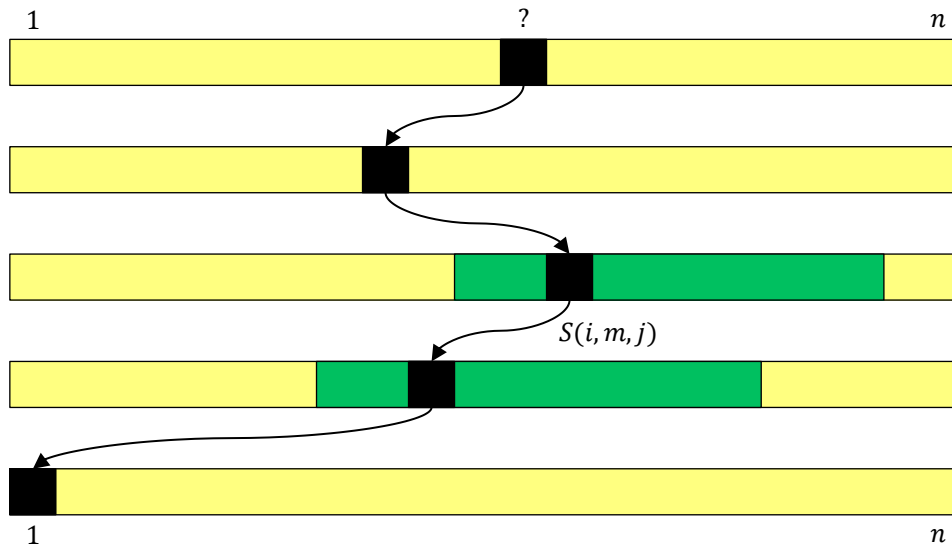
Thuật toán

Nếu ta thực hiện các phép tráo từ đầu đến cuối để suy ra trạng thái cuối cùng của bộ bài thì các kỹ thuật dùng mảng hay string khá chậm. Không phải không làm theo cách này được nhưng sẽ phải code những cấu trúc dữ liệu khó mà ta chưa học.

Vì k khá nhỏ, ta nghĩ đến việc không cần biết trạng thái của bộ bài cuối cùng thế nào, chỉ cần quan tâm k lá bài đầu tiên mang số mấy mà thôi. Ta trình bày cách dò lá bài đầu tiên trong trạng thái cuối, những lá bài khác tương tự...

- ✿ Một lá bài đứng đầu sau phép tráo thứ x thì nó sẽ phải đứng ở vị trí p_{x-1} nào đó trước phép tráo thứ x và vị trí này có thể tính được qua các tham số của phép tráo thứ x .
- ✿ Một lá bài muốn đứng ở vị trí $p[x-1]$ sau phép tráo thứ $x-1$ thì nó sẽ phải đứng ở vị trí p_{x-2} nào đó trước phép tráo thứ $x-1$ và vị trí này có thể tính được qua các tham số của phép tráo thứ $x-1$...
- ✿ Cuối cùng ta xác định được p_1 là vị trí của lá bài trước phép tráo thứ nhất tức là vị trí xuất phát của lá bài. Đó chính là số ghi trên lá bài cần tìm.

Xem hình vẽ



Có thể coi như việc tráo bài giống như hai thao tác rất phổ biến trong soạn thảo văn bản

Viết ra 1 dòng văn bản coi như dãy lá bài, một phép tráo bao gồm:

Cut (CTRL+X): Lấy ra một đoạn liên tiếp

Paste (CTRL+V): Chèn đoạn đã cắt vào một chỗ nào đó.

Việc của ta là viết một hàm

Undo(i, m, j, p): Trả về vị trí q mà sau phép tráo $S(i, m, j)$ lá bài ở vị trí đó sẽ về đúng vị trí thứ p . Hàm này không cần có một phép lặp hay đệ quy nào cả. Chỉ căn cứ vào giá trị p và các tham số i, m, j để tiến hành tính toán. Hàm này sẽ được thực hiện x lần từ phép tráo cuối cùng lên phép tráo đầu tiên để dò giá trị trên một lá bài

Thực chất là mô phỏng thao tác Undo (Ctrl+Z)

Nhận xét: thời gian thực hiện một phép Undo(i, m, j, p) không phụ thuộc vào n . Thời gian thực hiện toàn giải thuật cũng không phụ thuộc vào n , có thể chỉ ra rằng thời gian thực hiện giải thuật tỉ lệ thuận với tích $k * x$.