

Tổng quan đề thi

STT	Tên bài	Tên tệp bài làm	Đầu vào	Đầu ra	Điểm
1	Thuật toán sắp xếp	SAPXEP.*	SAPXEP.INP	SAPXEP.OUT	7
2	Xâu con không liên tiếp	SUBSTR.*	SUBSTR.INP	SUBSTR.OUT	7
3	Bẫy chuột	MOUTRAP.*	MOUTRAP.INP	MOUTRAP.OUT	6

Bài 1. Thuật toán sắp xếp

Sắp xếp là một trong những bài toán phổ biến nhất trong khoa học máy tính. An vừa được giới thiệu các thuật toán sắp xếp và biết rằng thuật toán sắp xếp nhanh (Quick Sort) có độ phức tạp trung bình là $O(N * \log N)$. Tuy nhiên An vẫn muốn sáng tạo một thuật toán sắp xếp mới và xem thử liệu nó có tốt hơn Quick Sort. Xét một mảng N số nguyên, mỗi phần tử có giá trị từ 1 đến N và mỗi giá trị chỉ xuất hiện một lần trong mảng. Thuật toán này sẽ sắp xếp mảng trên trong N bước như sau:

- Bước 1: phần tử có giá trị 1 sẽ được chuyển đến vị trí 1 bằng cách liên tiếp hoán đổi vị trí với phần tử liền kề với nó.
- Bước 2: phần tử có giá trị N sẽ được chuyển đến vị trí N bằng cách như bước 1.
- Bước 3: phần tử có giá trị 2 sẽ được chuyển đến vị trí 2 bằng cách tương tự.
- Bước 4: phần tử có giá trị $N - 1$ sẽ được chuyển đến vị trí $N - 1$ bằng cách tương tự.
- Và cứ thế tiếp tục đến bước N .

Tổng quát hơn, ở bước lẻ, An sẽ chọn phần tử có giá trị nhỏ nhất chưa được sắp xếp và di chuyển phần tử này đến vị trí đúng của nó. Ở bước chẵn An sẽ chọn phần tử có giá trị lớn nhất chưa được sắp xếp và di chuyển phần tử này đến vị trí đúng của nó.

Yêu cầu: Viết chương trình cho biết số lần hoán đổi vị trí các phần tử tại mỗi bước của thuật toán.

Dữ liệu: vào từ file **SAPXEP.INP** gồm:

- Dòng đầu tiên chứa một số nguyên N cho biết số lượng phần tử của mảng.
- Trong N dòng tiếp theo, dòng thứ i chứa một số nguyên a_i ($1 \leq a_i \leq N$) cho biết giá trị của phần tử thứ i trong mảng. Các giá trị a_i không lặp lại.

Kết quả: ghi ra file **SAPXEP.OUT** gồm N số nguyên, số thứ i cho biết số lần hoán đổi vị trí các phần tử ở bước thứ i của thuật toán. Mỗi số ghi trên một dòng.

Ví dụ:

INPUT	OUTPUT
7	5
5	2
2	1
6	2
7	1
3	1
1	0
4	

Ràng buộc:

- 50% test ứng với 50% số điểm của bài có $N \leq 1.000$;
- 20% test ứng với 20% số điểm của bài có $1000 < N \leq 10.000$;
- 30% test ứng với 30% số điểm của bài có $10.000 < N \leq 100.000$.

Bài 2. Xâu con không liên tiếp

Cho xâu s và n xâu t_i , hãy đếm xem có có bao nhiêu xâu t_i có nhiều hơn một cách chọn xâu con của s để được xâu t_i , xâu con trong bài này là xâu con có thể không liên tiếp.

Dữ liệu: vào từ file **SUBSTR.INP** gồm:

- Dòng đầu tiên chứa xâu s có độ dài m ($1 \leq m \leq 10^5$);
- Dòng thứ hai chứa số n ($1 \leq n \leq 10^5$)
- n dòng tiếp theo, mỗi dòng chứa một xâu t_i chỉ chứa những chữ cái in thường, gọi k là tổng độ dài của các xâu này ($1 \leq k \leq 5 \times 10^5$)

Kết quả ra: ghi ra file **SUBSTR.OUT** một số nguyên là số lượng xâu t_i thỏa mãn.

Ví dụ:

INPUT	OUTPUT
ababa	2
6	
aa	
aba	
abb	
baa	
abb	
xyz	

Giới hạn:

- 30% số test đầu tiên tương ứng với $1 \leq n, m, k \leq 100$
- 30% số test tiếp theo tương ứng với $100 \leq n, m, k \leq 1000$
- 40% số test còn lại không có ràng buộc gì thêm.

Bài 3. Bẫy chuột

Chú mèo máy Doreamon có một mê cung khổng lồ với n căn phòng được đánh số $1 \dots n$ và $n - 1$ lối đi sao cho tất cả các phòng đều có thể đi đến được với nhau. Thật không may, một con chuột đã lẻn vào mê cung. Doreamon cực kỳ sợ chuột nên đặt bẫy chuột trong phòng t . Rõ ràng chuột tránh được căn phòng có bẫy nên Doreamon phải nghĩ ra chiến lược tốt hơn để dụ chuột vào bẫy. Chuột liên tục chạy xung quanh và không bao giờ dừng lại, trừ khi nó không còn nơi nào để di chuyển. Doreamon cũng biết rằng con chuột để lại dấu chân ở mỗi lối đi mà nó đi qua. Con chuột sau đó sẽ không sử dụng lại lối đi ấy một lần nào nữa. Doreamon có thể dọn sạch sẽ lối đi để lại dấu chân hoặc chặn lối đi bằng đá. Bằng cách chặn các lối đi hoặc dọn sạch vết chân của chuột, anh ta muốn buộc con chuột chạy vào bẫy. Anh ấy muốn thực hiện điều này với số lần tác động (dọn sạch hoặc chặn đá) tối thiểu vì anh ấy cảm thấy rất khó chịu khi có con chuột.

Chúng ta có thể mô tả đây như một trò chơi dành cho hai người chơi. Con chuột cố gắng tối đa hóa số lần tác động của Doreamon. Doreamon cố gắng giành chiến thắng với số lần tác động tối thiểu. Người chơi đầu tiên là Doreamon. Đến lượt của mình, anh ta có thể dọn sạch một lối đi có vết chân chuột hoặc chặn một lối đi. Việc lối đi bị chặn có vết chân hay không không quan trọng. Anh ta không thể bỏ chặn một lối đi. Tuy nhiên, anh ta có thể chọn không làm gì cả. Lượt đi mà Doreamon quyết định không làm gì sẽ không được tính là lượt đi. Khi đến lượt chuột, chuột sẽ chọn một lối đi sạch sẽ không bị chặn và chạy sang phòng bên cạnh theo lối đi đó. Nếu không có lối đi như vậy dẫn từ phòng hiện tại của chuột, chuột sẽ không di chuyển.

Ban đầu, tất cả các lối đi đều sạch sẽ, chuột ở phòng m , bẫy ở phòng t và lượt đầu tiên dành cho Doreamon. Số lần thao tác (dọn sách hoặc chặn đá các lối đi) tối thiểu mà Doreamon cần là bao nhiêu nếu cả hai nhân vật chơi tối ưu (mục tiêu của chuột là tối đa hóa số lần di chuyển của Doreamon)?.

Dữ liệu: vào từ file **MOUTRAP.INP**:

- Dòng đầu tiên: chứa các số nguyên n, t và m cách nhau bằng dấu cách.
- $n - 1$ dòng tiếp theo, trong mỗi dòng ghi a_i và b_i , cách nhau bằng dấu cách, biểu thị lối đi giữa phòng a_i và b_i .

Kết quả: in ra file **MOUTRAP.OUT**: In ra số thao tác ít nhất của Doreamon để bẫy được chuột.

Ví dụ:

MOUTRAP.INP	MOUTRAP.OUT	Giải thích
10 1 4 1 2 2 3 2 4 3 9 3 5 4 7 4 6 6 8 7 10	4	• Doreamon chặn lối đi giữa phòng 4 và 7. • Chuột di chuyển đến phòng 6. Lối đi giữa phòng 4 và 6 có dấu chân. • Doreamon chặn lối đi giữa phòng 6 và 8. • Chuột không thể di chuyển. • Doreamon dọn dẹp lối đi giữa phòng 4 và 6. • Chuột di chuyển đến phòng 4. Lối đi giữa phòng 4 và 6 có dấu chân. • Doreamon chặn lối đi giữa phòng 2 và 3. • Chuột di chuyển đến phòng 2. Lối đi giữa phòng 2 và 4 có dấu chân. • Doreamon không làm gì cả. • Chuột chỉ có thể di chuyển đến phòng 1 và bị mắc bẫy. Doreamon thực hiện 4 nước đi.

Giới hạn: $1 \leq n, t, m \leq 1.000.000$

Ràng buộc:

- 20% số test $n \leq 10$
- 25% số test tiếp theo đảm bảo có lối đi trực tiếp nối phòng m với phòng t .
- 20% số test tiếp theo $n \leq 1000$
- 35% số test cuối cùng không có ràng buộc gì.

-----HẾT-----

- Họ tên thí sinh:.....

- Số báo danh:..... Phòng thi:.....

Người ra đề: Vương Thành Trung

SĐT: 0839051984