

ĐỀ CHÍNH THỨC

(Đề thi có 04 trang)

ĐỀ THI MÔN: TIN HỌC

Thời gian: 180 phút (không kể thời gian giao đề)

Ngày thi: 22/02/2025

TỔNG QUAN VỀ ĐỀ THI

TT	Tên bài	Tên CT	Dữ liệu vào	Dữ liệu ra	Điểm
Bài 1	Thời gian chờ	WAIT.*	WAIT.INP	WAIT.OUT	5
Bài 2	Bộ số bằng nhau	EQUAL*	EQUAL.INP	EQUAL.OUT	5
Bài 3	Xoay bảng	ROTA.*	ROTA.INP	ROTA.OUT	5
Bài 4	Cây đồ thị	TREE.*	TREE.INP	TREE.OUT	5

Phần mở rộng * là CPP hoặc PY tùy theo môi trường lập trình C++, Python.

Hãy lập trình giải những bài toán sau:

BÀI 1. Thời gian chờ

Hiếu là một hướng dẫn viên hỗ trợ du khách tham quan lễ hội ẩm thực sắp diễn ra tại Hội An. Hiếu nhận được thông tin từ ban tổ chức về n cửa hàng, cửa hàng thứ i ($1 \leq i \leq n$) sẽ mở cửa đón khách từ thời điểm L_i và không nhận khách từ thời điểm R_i trở đi. Có m du khách đã đặt lịch trình tham gia lễ hội. Du khách thứ j ($1 \leq j \leq m$) sẽ tới vào thời điểm k_j và dự kiến sẽ vào cửa hàng bất kỳ nào đó đang mở đón khách. Trường hợp không có cửa hàng nào mở, du khách sẽ đợi và vào cửa hàng mở cửa đón khách sớm nhất có thể.

Yêu cầu: Hãy giúp Hiếu xác định thời gian chờ đợi của từng du khách để vào được cửa hàng. Biết thời gian di chuyển không đáng kể.

Dữ liệu: vào từ tệp **WAIT.INP**

- Dòng đầu chứa 2 số nguyên dương n, m ($n, m \leq 10^5$)
- n dòng tiếp theo, dòng thứ i chứa 2 số nguyên dương L_i, R_i ($L_i < R_i \leq 10^9$)
- m dòng cuối cùng, dòng thứ j chứa một số nguyên dương k_j ($k_j \leq 10^9$) xác định thời điểm tới lễ hội của du khách thứ j

Kết quả: ghi ra tệp **WAIT.OUT** gồm m dòng, dòng thứ j chứa thông tin trả lời là thời gian chờ đợi để được vào cửa hàng của du khách thứ j kể từ khi có mặt tại lễ hội. Trường hợp du khách không được vào một cửa hàng nào đưa ra -1 .

Ràng buộc:

- 40% số điểm với $m, n \leq 10^3$; $R_i \leq 10^6 \forall i = 1, 2, \dots, n$. và tại một thời điểm, không có 2 cửa hàng nào cùng mở cửa (hay với $i \neq j$, ta có $L_i > R_j$ hoặc $L_j > R_i$). $L_i < R_i \leq 10^6 \forall i = 1, 2, \dots, n$)
- 30% số điểm với $10^3 \leq m, n \leq 10^6$ và tại một thời điểm, không có 2 cửa hàng nào cùng mở cửa (hay với $i \neq j$, ta có $L_i > R_j$ hoặc $L_j > R_i$). $L_i < R_i \leq 10^6 \forall i = 1, 2, \dots, n$
- 30% số điểm còn lại không có ràng buộc bổ sung.

Ví dụ:

WAIT.INP	WAIT.OUT	Giải thích
3 4	0	- Người đầu tiên tới, vào luôn cửa hàng 1, không cần chờ đợi.
1 5	2	- Người thứ 2 tới, đúng lúc cửa hàng 1 đóng cửa, cần đợi vào cửa hàng mở gần nhất là cửa hàng 2.
7 9	-1	- Người thứ 3 tới muộn, tất cả các cửa hàng đã đóng cửa.
10 13	1	- Người thứ 4 tới, đợi cửa hàng 2 mở cửa.
3		
5		
14		
6		

BÀI 2. Bộ số bằng nhau

Trong lập trình, lệnh Random cho phép người lập trình tạo ra một giá trị ngẫu nhiên. Bạn Hiếu sử dụng lệnh này nhiều lần để tạo ra dãy N số nguyên $a_1, a_2, \dots, a_N$. Với M giá trị bằng nhau: $a_{j_1}, a_{j_2}, \dots, a_{j_M}$ ($1 \leq j_1 < j_2 < \dots < j_M \leq N$), Hiếu nhận thấy chúng có thể tạo thành các bộ nhị (bộ hai phần tử bằng nhau), bộ tam (bộ ba phần tử bằng nhau), bộ tứ (bộ bốn phần tử bằng nhau), ..., bộ K (bộ K phần tử bằng nhau).

Ví dụ: $a_{j_1} = a_{j_2}$, với $j_1 < j_2$ sẽ tạo thành một bộ nhị.

$a_{j_1} = a_{j_2} = a_{j_3}$, với $j_1 < j_2 < j_3$ sẽ tạo thành một bộ tam.

$a_{j_1} = a_{j_2} = \dots = a_{j_K}$, với $j_1 < j_2 < \dots < j_K$ sẽ tạo thành một bộ K .

Yêu cầu: Với số nguyên dương K cho trước, hãy giúp bạn Hiếu xác định số lượng các bộ K .

Dữ liệu: Vào từ tệp văn bản **EQUAL.INP** gồm:

- Dòng 1: Ghi số nguyên dương N ($N \leq 10^6$) và số nguyên dương T là số lượng bộ test ($T \leq 200$).
- Dòng 2: Ghi N số nguyên $a_1, a_2, \dots, a_N$ ($|a_i| \leq 10^6, 1 \leq i \leq N$).
- T dòng tiếp theo, mỗi dòng ghi một số nguyên dương K - cho biết cần tìm số lượng các bộ K ($2 \leq K \leq N$) có thể tạo thành.

Các số trên một dòng cách nhau bởi một dấu cách.

Kết quả: Ghi ra tệp văn bản **EQUAL.OUT** gồm T dòng tương ứng với T test, với mỗi test ghi kết quả là số lượng các bộ K . Vì số lượng các bộ thỏa mãn có thể rất lớn nên chỉ cần in ra phần dư cho $10^9 + 7$.

Ví dụ:

EQUAL.INP	EQUAL.OUT	Giải thích
6 2	4	Các giá trị bằng nhau:
4 5 4 6 6 4	0	$(a_1 = a_3 = a_6); (a_4 = a_5)$
2		Khi $K = 2$, số lượng các bộ nhị là 4:
4		$(a_1, a_3); (a_1, a_6); (a_3, a_6); (a_4, a_5)$
		Khi $K = 4$, số lượng các bộ tứ là 0.

Ràng buộc:

- 20% số test với $K \leq N \leq 20; T \leq 10; 0 \leq a_i \leq 10^6$;
- 40% số test với $N \leq 2 \cdot 10^5; T = 1; K = 2; 0 \leq a_i \leq 10^6$;
- 40% số test với ràng buộc gốc.

BÀI 3. Xoay bảng

Cho ma trận kích thước $N \times N$. Các hàng được đánh số từ 1 tới N , các cột đánh số từ 1 tới N . Tại ô (i, j) ở hàng i , cột j chứa một số nguyên a_{ij} ($|a_{ij}| \leq 10^6$; $1 \leq i, j \leq N$). Thực hiện lần lượt Q truy vấn, mỗi truy vấn thuộc một trong ba loại sau:

- $R\ X$: Xoay ma trận theo chiều quay kim đồng hồ 90° X lần.
- $L\ Y$: Xoay ma trận ngược chiều quay kim đồng hồ 90° Y lần.
- $3\ K$: Tìm đường đi có tổng lớn nhất từ một ô bất kì ở cột 1 đến một ô bất kì ở cột K .

Biết rằng từ ô (i, j) có thể di chuyển đến ô $(i - 1, j + 1)$, $(i, j + 1)$ và $(i + 1, j + 1)$.
Đưa ra tổng lớn nhất.

Dữ liệu: Vào từ tệp văn bản **ROTA.INP** gồm:

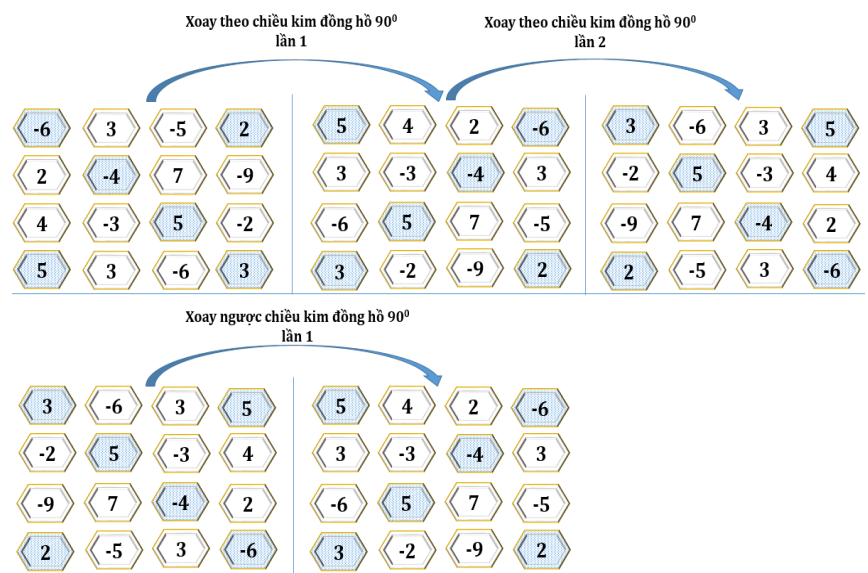
- Dòng 1: Ghi số nguyên dương N và Q ($N, Q \leq 50$).
- N dòng tiếp theo, mỗi dòng ghi N số nguyên.
- Q dòng tiếp theo, mỗi dòng chứa một trong ba loại truy vấn trên.

Dữ liệu vào có ít nhất một truy vấn dạng $3\ K$. Các số trên một dòng cách nhau bởi một dấu cách.

Kết quả: Ghi ra tệp văn bản **ROTA.OUT** gồm nhiều dòng, mỗi dòng ghi kết quả của một truy vấn dạng $3\ K$.

Ví dụ:

ROTA.INP	ROTA.OUT
4 5	18
-6 3 -5 2	16
2 -4 7 -9	
4 -3 5 -2	
5 3 -6 3	
R 2	
L 1	
3 4	
L 1	
3 4	



Ràng buộc:

- $0 \leq X, Y \leq 50$; $1 \leq K \leq N \leq 50$;
- Có 20% test chỉ có truy vấn loại 3.
- Có 30% test với X và Y chia hết cho 4;
- Có 50% test với ràng buộc gốc.

BÀI 4. Cây đồ thị

Cho một cây gồm n đỉnh (cây là một đồ thị liên thông n đỉnh $n - 1$ cạnh). Có m truy vấn và số nguyên k . Mỗi truy vấn bạn được cho s đỉnh, bạn cần sử dụng các cạnh của cây ban đầu để đảm bảo mọi cặp đỉnh trong s đỉnh đã cho đều đến được nhau thông qua các cạnh đã lựa chọn.

Yêu cầu: Sau khi thực hiện m truy vấn hãy in ra các cạnh của cây ban đầu được sử dụng không ít hơn k lần.

Dữ liệu vào: từ tệp văn bản **TREE.INP** gồm:

- Dòng đầu tiên ghi ba số nguyên n, m và k .
- $n - 1$ dòng tiếp theo dòng thứ i ghi hai số nguyên u_i, v_i ($1 \leq u_i, v_i \leq n, u_i \neq v_i$) thể hiện cạnh thứ i nối hai đỉnh u_i, v_i ;
- m dòng tiếp theo, dòng thứ i số đầu tiên ghi số s_i là số đỉnh của truy vấn thứ i , tiếp theo s_i số $u_{i1}, u_{i2}, \dots, u_{is_i}$ là các đỉnh của truy vấn thứ i .

Dữ liệu của bài đảm bảo: $2 \leq s_i \leq n \leq 100000, S = \sum_{i=1}^m s_i \leq 100000, 1 \leq k \leq m \leq 50000$.

Kết quả: ghi ra tệp văn bản **TREE.OUT** gồm:

- Dòng đầu ghi số nguyên r là số lượng các cạnh thỏa mãn yêu cầu bài toán
- Dòng thứ 2 ghi r số nguyên là chỉ số của các cạnh thỏa mãn yêu cầu của bài toán theo thứ tự tăng dần.

Dữ liệu đảm bảo luôn có đáp án $r > 0$.

Ví dụ:

TREE.INP	TREE.OUT	Hình minh họa đồ thị
6 3 2 1 3 2 3 3 4 6 4 4 5 4 1 3 2 5 2 6 3 2 3 2	2 2 3	

Giải thích: Với truy vấn thứ nhất cần dùng các cạnh thứ 1,2,3 và 5. Với truy vấn thứ 2 cần dùng các cạnh thứ 3, 4. Với truy vấn thứ 3 cần dùng cạnh thứ 2. Vậy các cạnh được sử dụng không ít hơn 2 là cạnh thứ 2 và cạnh thứ 3.

Ràng buộc:

- Có 20% số test tương ứng với 20% số điểm của bài đồ thị thỏa mãn cạnh thứ i ($i = 1, 2, \dots, n - 1$) có $u_i = i, v_i = i + 1; 2 \leq n \leq 5000$;
- Có 20% số test tương ứng với 20% số điểm của bài thỏa mãn $n \leq 10000, m \leq 2000$;
- Có 30% số test tương ứng với 30% số điểm của bài có $k = m$ và $s_i = 2 \forall i = 1, 2, \dots, m$;
- Ràng buộc 4: có 30% số test tương ứng với 30% số điểm của bài không có ràng buộc gì thêm.

----- Hết -----

Họ tên :SBD :