

CONTEST 09*Thời gian: 180'***TỔNG QUAN ĐỀ THI**

	Tiêu đề	File chương trình	File dữ liệu	File kết quả
Câu 1	Nước	WATER.*	WATER.INP	WATER.OUT
Câu 2	Khám phá	COUNT.*	COUNT.INP	COUNT.OUT
Câu 3	Làm việc nhóm	WORKING.*	WORKING.INP	WORKING.OUT

Dấu * được thay bởi PAS hoặc CPP tương ứng với ngôn ngữ lập trình Pascal hoặc C++.

Hãy lập trình giải các câu sau:

Câu 1 (6 điểm). Nước

An có n cái cốc được gắn nhãn lần lượt $1, 2, \dots, n$ với dung tích không hạn chế và trong mỗi cốc đều có nước. An muốn uống hết số nước ở tất cả các cốc nhưng chỉ muốn uống ở đúng k cốc. Để có thể làm được như vậy, cậu ta phải rót nước từ cốc này sang cốc khác để chỉ còn đúng k cốc có nước. Vấn đề đặt ra với An là chọn cốc nào để rót nước sang cốc nào, bởi vì khoảng cách giữa các cốc là không bằng nhau. Cụ thể, An sẽ mất một lượng thời gian khi rót nước từ cốc i sang cốc j gọi nó là C_{ij} . Em hãy giúp An tìm thứ tự rót nước từ cốc này sang cốc khác để tổng thời gian cậu ấy bỏ ra là ít nhất. Thỏa mãn điều kiện, An có thể uống hết lượng nước chỉ trong k cốc.

Dữ liệu vào file văn bản WATER.INP:

- Dòng 1 chứa hai số nguyên n, k ($1 \leq k \leq n \leq 20$);
- N dòng sau, dòng thứ i chứa n số nguyên C_{ij} ($0 \leq C_{ij} \leq 10^5$) là lượng thời gian cần dùng khi rót nước từ cốc i sang cốc j . Nếu $i = j$ thì $C_{ij} = 0$.

Kết quả:

Ghi ra file văn bản **WATER.OUT** là một số duy nhất là tổng thời gian ít nhất cần sử dụng.

Ví dụ:

WATER.INP	WATER.OUT	Giải thích
4 2 0 8 49 23 7 0 30 22 44 28 0 23 9 40 15 0	16	Rót nước từ cốc $2 \rightarrow 1$ (tốn 7), sau đó rót từ $4 \rightarrow 1$ (tốn 9). Nước cuối cùng còn lại ở cốc 1 và cốc 3. Tổng thời gian rót nước là: $7+9=16$

Ràng buộc:

- (1) Có 40% số test tương ứng với 40% số điểm có $1 \leq n \leq 10$;
- (2) Có 60% số test tương ứng với 60% số điểm có $1 \leq n \leq 20$.

Câu 2 (7,0 điểm). Khám phá

Tèo rất đam mê với việc khám phá vũ trụ, nên cậu đăng ký tham gia Tổ chức Thiên văn của trái đất, Tổ chức này đang đánh giá khả năng tồn tại sự sống của các hành tinh trong

giải Ngân hà. Có N ($1 \leq N \leq 10^6$) loại hành tinh đã được nghiên cứu, khả năng tồn tại sự sống tại các loại hành tinh là các số nguyên a_1, a_2, \dots, a_N ($0 \leq a_i \leq 10^9$).

Tổ chức đã tìm ra một hệ số sinh tồn M ($1 \leq M \leq 3000$), là hệ số sinh tồn hoàn hảo cho con người. Nếu kết hợp 3 hành tinh lại, mà có **tích** khả năng tồn tại sự sống của chúng chia hết cho M thì là có thể tạo ra một hành tinh mới gần giống trái đất.

Tổ chức Thiên văn đưa ra bài kiểm tra cho Tèo hãy đếm số cách kết hợp loại 3 hành tinh để con người có thể sinh sống. Hay nói cách khác, đếm số bộ 3 số (i, j, k) sao cho $a_i * a_j * a_k$ mà chia hết cho M . Chú ý: các số i, j, k có thể bằng nhau; bộ số (i_1, j_1, k_1) giống bộ số (i_2, j_2, k_2) khi $i_1 = i_2, j_1 = j_2, k_1 = k_2$.

Dữ liệu

Vào từ file văn bản **COUNT.INP**:

- Dòng đầu ghi số hai số N, M ($1 \leq N \leq 10^6, 1 \leq M \leq 3000$).
- Dòng tiếp theo ghi N số a_1, a_2, \dots, a_N ($0 \leq a_i \leq 10^9$).

Kết quả

Ghi ra file văn bản **COUNT.OUT** là số cách kết hợp các loại hành tinh để tồn tại sự sống cho con người.

Ví dụ:

COUNT.INP	COUNT.OUT	Giải thích
2 5 1 5	7	Với ví dụ 1: có 7 cách kết hợp tương ứng với bộ các chỉ số: (1, 1, 2), (1, 2, 1), (1, 2, 2), (2, 1, 1), (2, 1, 2), (2, 2, 1), (2, 2, 2). Khi đó các tích $a_i * a_j * a_k$ tương ứng đều chia hết cho 5
10 3 1 2 3 4 5 6 7 8 9 10	657	

Ràng buộc:

- (1) Có 20% số test ứng với 20% số điểm thoả mãn $1 \leq N \leq 200$;
- (2) Có 20% số test khác ứng với 20% số điểm thoả mãn $200 < N \leq 2000$;
- (3) Có 20% số test khác ứng với 20% số điểm thoả mãn $1 \leq M \leq 200$;
- (4) Có 40% số test còn lại ứng với 40% số điểm không có ràng buộc gì thêm.

Câu 3 (7,0 điểm). Làm việc nhóm

Trong công ty XYZ có N nhân viên rất xuất sắc. Tuy nhiên do tất cả đều quá giỏi và quá tự tin, nên cứ khi nào 2 nhân viên cùng làm việc với nhau thì hiệu suất coi như bằng 0, vì họ tốn thời gian vào việc tranh cãi và không quyết định được công việc gì. Nhân viên thứ i có giờ làm việc là một khoảng thời gian liên tiếp từ thời điểm a_i đến b_i ($a_i < b_i$). Giờ làm việc của mỗi nhân viên không thể thay đổi do đặc điểm công việc mà họ đảm trách và tính kỳ quặc của họ. Do các khoảng thời gian này không giống nhau hoàn toàn, có những lúc chỉ có một nhân viên làm việc, lúc đó họ làm việc rất hiệu quả. Giám đốc muốn giữ lại một số nhân viên sao cho tổng thời gian làm việc hiệu quả là lớn nhất.

Dữ liệu vào: Từ tệp **WORKING.INP**

- Dòng đầu tiên ghi N là số nhân viên ($1 \leq N \leq 10^5$)
- N dòng tiếp theo mỗi dòng ghi hai số nguyên dương a_i và b_i là thời điểm bắt đầu và kết thúc giờ làm việc của nhân viên i ($0 \leq a_i < b_i \leq 10^9$)

Kết quả ra: Ghi ra tệp **WORKING.OUT**

- Một số nguyên duy nhất là tổng thời gian làm việc hiệu quả lớn nhất có thể.

Ví dụ:

WORKING.INP	WORKING.OUT	Giải thích
7 100 150 200 300 900 1000 100 1000 900 1800 1900 2000 1800 2000	1800	Các nhân viên được giữ lại có thứ tự là: 4, 5, 7. Tương ứng với các khoảng thời gian: [100,1000],[900,1800], [1800,2000]

Ràng buộc:

(1) Có 20% số test ứng với 20% số điểm có khoảng thời gian làm việc của từng người là rời nhau;

(2) Có 50% số test ứng với 50% số điểm có $N \leq 10^4$;

(3) Có 30% test ứng với 30% số điểm có $10^4 < N \leq 10^5$.

..... **HẾT**

- Thí sinh không sử dụng tài liệu.
- Cán bộ coi thi không giải thích gì thêm.