

Thời gian làm bài: **180 phút**, không kể thời gian giao đề

ĐỀ THI MINH HỌA

(Đề thi này có 04 trang)

**TỔNG QUAN ĐỀ THI**

Bài	Tên bài	Tệp chương trình	Tệp dữ liệu	Tệp kết quả	Bộ nhớ	Thời gian / test	Điểm
1	Hợp kim	alloy.*	alloy.inp	alloy.out	1024 MB	1 giây	6
2	Tổng dư	modsum.*	modsum.inp	modsum.out	1024 MB	1 giây	6
3	Đếm xâu	cstring.*	cstring.inp	cstring.out	1024 MB	1 giây	5
4	Chi phí	cost.*	cost.inp	cost.out	1024 MB	1 giây	3

Dấu \* được thay thế bởi CPP hoặc PY của ngôn ngữ lập trình được sử dụng tương ứng là C++ hoặc Python.

**Hãy lập trình giải các bài toán sau:**

**Bài 1. Hợp kim (6 điểm)**

Eldoria là một vương quốc cổ kính, được bao phủ bởi những cánh rừng rậm rạp, những dòng sông trong vắt và những ngọn núi hùng vĩ. Nơi đây, phép thuật hòa quyện với tự nhiên, tạo nên một thế giới kỳ ảo và đầy màu sắc. Người dân Eldoria sống trong hòa bình và hạnh phúc, họ biết ơn những điều kỳ diệu mà đất mẹ ban tặng.

Tuy nhiên, một thế lực hắc ám từ đáy vực sâu đã trỗi dậy. Đó là Morgoth, một chúa quỷ cổ đại, hắn ta cùng với lũ quái vật đáng sợ đã xâm chiếm những vùng đất lân cận và giờ đây, hắn ta nhắm đến Eldoria. Morgoth muốn biến Eldoria thành một vương quốc của bóng tối, nơi mà nỗi sợ hãi và đau khổ ngự trị.

Trước mối đe dọa khủng khiếp này, người dân Eldoria vô cùng lo lắng. Họ đã tìm đến Gorim, một chàng thợ rèn người lùn nổi tiếng với tài năng chế tạo những vũ khí phép thuật. Với đôi bàn tay khéo léo và một trái tim nhiệt huyết, Gorim đã nhận lời giúp đỡ người dân Eldoria. Anh ta biết rằng để đánh bại Morgoth và lũ quái vật của hắn, cần phải có những vũ khí đặc biệt, được rèn từ hai loại nguyên liệu quý hiếm là: “*trinh thể lửa*”, được sinh ra từ lòng đất, tượng trưng cho sự dũng cảm và sức mạnh; “*bột gió*”, được tạo thành từ những cơn lốc xoáy, lại mang đến sự nhanh nhẹn và linh hoạt.

Gorim đã khám phá ra rằng việc kết hợp “*trinh thể lửa*” và “*bột gió*” theo những tỷ lệ khác nhau sẽ tạo ra những vũ khí phép thuật vô cùng mạnh mẽ. Để tìm ra những tỷ lệ này thì ứng với mỗi đoạn số  $[a; b]$  được cho trước Gorim phải thực hiện hai phép toán:

- Tìm số lượng số nguyên tố*: đầu tiên, Gorim cần tìm số lượng số nguyên tố nằm trong đoạn số  $[a; b]$ . Số lượng này sẽ là số lượng “*trinh thể lửa*” cần phải có.
- Tính trung bình cộng các số nguyên tố*: tiếp theo, Gorim cần tính trung bình cộng của tất cả các số nguyên tố trong đoạn  $[a; b]$ . Giá trị trung bình cộng này chính là số lượng cần có của nguyên liệu “*bột gió*”.

**Yêu cầu:** Bạn hãy lập trình giúp Gorim hoàn thành công việc tính toán phức tạp nói trên.

**Đầu vào:** Vào từ file văn bản **alloy.inp**:

- Dòng đầu tiên chứa số nguyên dương  $n$  ( $1 \leq n \leq 10^4$ ) cho biết số lượng vũ khí Gorim phải rèn.
- $n$  dòng tiếp theo, mỗi dòng ghi hai số nguyên dương  $a, b$  ( $1 \leq a < b \leq 10^6$ ).

**Kết quả:** Ghi ra file văn bản **alloy.out**: Gồm  $n$  dòng, mỗi dòng chứa hai giá trị phân biệt nhau bởi một khoảng trắng, ứng với số lượng “*tinh thể lửa*” và “*bột gió*” cần thiết cho mỗi loại vũ khí. Trong đó giá trị “*bột gió*” được làm tròn đến chữ số thập phân thứ hai.

Các số được viết trên một dòng của file dữ liệu vào/ra được ghi cách nhau bởi dấu cách.

**Ví dụ:**

alloy.inp	alloy.out
2	4 4.25
1 10	2 4.00
3 5	

**Ràng buộc:**

- Có 60% số test tương ứng với 60% số điểm của bài có  $1 \leq n \leq 10$ ;  $1 \leq a < b \leq 10^4$ ;
- 40% số test còn lại tương ứng với 40% số điểm của bài không có thêm ràng buộc nào.

## Bài 2. Tổng dư (6 điểm)

Hoán vị là một cách sắp xếp các phần tử của một tập hợp theo một thứ tự nhất định. Ví dụ, các hoán vị của tập hợp  $\{1, 2, 3\}$  là:  $\{1, 2, 3\}, \{1, 3, 2\}, \{2, 1, 3\}, \{2, 3, 1\}, \{3, 1, 2\}, \{3, 2, 1\}$ .

Khi chia một số  $a$  cho một số  $b$ , số dư là phần còn lại sau khi đã thực hiện phép chia hết nhiều lần nhất có thể. Ví dụ, khi chia 7 cho 3, ta được thương là 2 và số dư là 1.

Cho một số nguyên dương  $N$ . Ta cần tìm một hoán vị  $\{P_1, P_2, \dots, P_N\}$  của tập hợp  $\{1, 2, \dots, N\}$ . Với mỗi giá trị  $i$  từ 1 đến  $N$ , ta tính  $M_i$  là số dư khi chia  $i$  cho  $P_i$ .

**Yêu cầu:** Hãy tìm giá trị lớn nhất có thể của tổng  $M_1 + M_2 + \dots + M_N$ .

**Dữ liệu:** Vào từ file văn bản **modsum.inp**: Gồm một dòng duy nhất chứa số nguyên  $N$  ( $1 \leq N \leq 10^9$ ).

**Kết quả:** Ghi ra file văn bản **modsum.out** giá trị lớn nhất có thể của tổng  $M_1 + M_2 + \dots + M_N$ .

**Ví dụ:**

modsum.inp	modsum.out
3	3
4	6

**Ràng buộc:**

- Có 40% số test tương ứng 40% số điểm của bài có  $1 \leq N \leq 10$ ;
- Có 30% số test tương ứng 30% số điểm của bài có  $10 < N \leq 10^7$ ;
- 30% số test còn lại tương ứng với 30% số điểm của bài không có thêm ràng buộc nào.

### Bài 3. Đếm xâu (5 điểm)

Cho hai xâu kí tự  $S$  và  $T$ . Đếm số lần xuất hiện của xâu  $S$  trong xâu  $T$ . Các ký tự trong xâu  $S$  và  $T$  là các chữ cái la tinh bao gồm các chữ cái in hoa:  $(A, B, C, \dots, Z)$ , in thường  $(a, b, c, \dots, z)$ .

**Dữ liệu:** Vào từ file văn bản `cstring.inp`:

- Dòng thứ nhất chứa xâu  $S$  (độ dài xâu  $S$  tối đa là  $10^6$ );
- Dòng thứ hai chứa xâu  $T$  (độ dài xâu  $T$  tối đa là  $10^6$ ).

**Kết quả:** Ghi ra file văn bản `cstring.out`: Một số nguyên là số lần xâu  $S$  xuất hiện trong xâu  $T$ .

**Ví dụ:**

<code>cstring.inp</code>	<code>cstring.out</code>
ab	3
abccabccab	

**Ràng buộc:**

- Có 10% số test tương ứng 10% số điểm có độ dài xâu  $S$  là 1;
- Có 30% số test tương ứng 30% số điểm có độ dài hai xâu  $S$  và  $T$  tối đa là  $10^3$ ;
- 60% số test còn lại tương ứng 60% số điểm có độ dài hai xâu  $S$  và  $T$  tối đa là  $10^6$ .

### Bài 4. Chi phí (3 điểm)

Chuẩn bị vào năm học mới, một trường học muốn sử dụng dịch vụ đưa đón trẻ đến trường. Nhà trường vẽ sơ đồ có  $n$  điểm đón. Các điểm đón được thông với nhau bởi  $n - 1$  con đường. Con đường giữa hai điểm đón  $x_i$  và  $y_i$  có độ dài  $w_i$  (mét). Mỗi điểm đón có  $t_i$  học sinh. Nhà trường dự kiến số tiền phải chi trả cho dịch vụ đưa đón trẻ là  $k$  (đồng).

Được biết chi phí để vận chuyển một học sinh đi trên đoạn đường 1 mét là 1 đồng.

Công ty cung cấp dịch vụ đưa đón trẻ muốn tính chi phí đưa đón toàn bộ học sinh, để đánh giá xem công ty có đủ khả năng ký hợp đồng với nhà trường hay không. Có 3 mức độ đánh giá:

- + “NO” – mức độ ứng với  $k$  nhỏ hơn trường hợp chi phí ít nhất.
- + “YES” – mức độ ứng với  $k$  lớn hơn hoặc bằng trường hợp chi phí ít nhất và  $k$  nhỏ hơn hoặc bằng trường hợp chi phí lớn nhất.
- + “GOOD” – mức độ ứng với  $k$  lớn hơn trường hợp chi phí lớn nhất.

**Dữ liệu:** Vào từ file văn bản `cost.inp`:

- Dòng đầu chứa số nguyên dương  $n$  ( $1 \leq n \leq 10^5$ ) - số điểm đón, và số nguyên dương  $k$  ( $1 \leq k \leq 10^{18}$ );
- $n - 1$  dòng tiếp theo, dòng thứ  $i$  chứa 3 số nguyên dương  $x_i, y_i, w_i$  xác định điểm đón thứ  $i$  ( $1 \leq x_i, y_i \leq n$ ;  $1 \leq w_i \leq 10^4$ );
- Dòng cuối chứa  $n$  số nguyên  $t_1, t_2, \dots, t_n$  ( $t_i \leq 10^4$ ) là số lượng học sinh cần đón của từng điểm theo thứ tự tăng dần từ 1 đến  $n$ .

**Kết quả:** Ghi ra file văn bản **cost.out**: Trên một dòng, gồm hai số nguyên là chi phí nhỏ nhất, chi phí lớn nhất và tiếp đó là mức độ đánh giá của công ty (NO, YES, GOOD).

Các số, từ được viết trên một dòng của file dữ liệu vào/ra được ghi cách nhau bởi dấu cách.

**Ví dụ:**

cost.inp	cost.out
3 20 1 2 5 2 3 6 3 1 2	27 39 NO
5 39 2 3 2 1 3 3 2 4 1 4 5 2 1 2 3 2 1	16 39 YES
4 200 2 1 5 3 2 10 2 4 1 1 5 2 9	33 164 GOOD

**Ràng buộc:**

- Có 60% số test tương ứng 60% số điểm có  $n \leq 10^3$ ;
- 40% số test còn lại tương ứng 40% số điểm có  $n \leq 10^5$ .

----- HẾT -----

- Thí sinh không được sử dụng tài liệu;

- Giám thị không giải thích gì thêm.

Họ và tên thí sinh: ..... Số báo danh: .....