

Mục lục

| | |
|-------------------------------------|---|
| Tìm số vắng mặt — missing | 1 |
| Tô màu lưới — COLORTAB | 3 |
| Chuỗi từ ghép — DECODE | 5 |

Nộp bài tại: bkict.org:50002

Username: họ tên của mình viết rút gọn. Ví dụ: Nguyễn Hoàng Hải Minh \rightarrow *minhnhh*

Password như username.

Bài A. Tìm số vắng mặt

| | |
|--------------------|----------|
| File dữ liệu vào: | Gọi hàm |
| File kết quả: | Trả về |
| Hạn chế thời gian: | 0.1 giây |
| Hạn chế bộ nhớ: | 256 MB |

Alice và Bob rủ nhau chơi trò chơi tìm các số vắng mặt sau đây. Từ các phần tử của dãy số $1, 2, \dots, n$ ($2 \leq n \leq 1000$), Alice xây dựng tập X gồm m số a_1, a_2, \dots, a_m với $a_1 < a_2 < \dots < a_m$. Nhiệm vụ của Bob là tìm ra $n - m$ số trong dãy $1, 2, \dots, n$ không có mặt trong tập các số X mà Alice đang giữ, bằng cách yêu cầu Alice trả lời câu hỏi có dạng sau đây:

“Bít thứ j trong biểu diễn nhị phân của số a_i là bao nhiêu?”

Câu trả lời của Alice cho câu hỏi sẽ là 0 hoặc 1. Lưu ý là các bit của một số nguyên k được đánh số từ 0 đến $\lfloor \log_2 k \rfloor$, từ phải qua trái.

Yêu cầu: Hãy giúp Bob tìm ra $n - m$ số không có mặt trong tập gồm m số mà Alice đang giữ với số lượng câu hỏi cần yêu cầu Alice trả lời càng ít càng tốt.

Chương trình của bạn phải sử dụng một thư viện riêng. Thư viện bao gồm các file sau: `missinglib.h` (cho C++). Trong chương trình của bạn các thư viện này cần được khai báo ở đầu chương trình:

```
#include "missinglib.h"
```

Thư viện cung cấp các hàm sau:

- Hàm khởi tạo trò chơi

```
int get_n():
```

```
int get_m():
```

Chương trình của bạn phải gọi các hàm này để khởi tạo trò chơi và lấy giá trị cho 2 biến n và m .

- Hàm thực hiện truy vấn

```
int ask(int i, int j);
```

Hàm này trả về 1 nếu bít thứ j của số a_i bằng 1 và trả về 0 nếu ngược lại.

- Hàm trả lời câu hỏi

```
void guess(vector<int> res);
```

Để kết thúc, chương trình của bạn cần gọi hàm này với `res` là kiểu vector trở đến $n - m$ số tìm được đã được sắp xếp theo thứ tự tăng dần của giá trị. Sau khi gọi hàm này chương trình sẽ tự động kết thúc. Số lượng câu hỏi của chương trình của bạn sẽ bằng tổng số lần gọi hàm $ask(i, j)$.

Lưu ý: Mỗi hàm `get_n`, `get_m` và `guess` chỉ được gọi một lần duy nhất.

Bạn có thể xem các file được cung cấp trên hệ thống để hiểu rõ hơn về cách tương tác với hệ thống.

Ví dụ

| Gọi hàm | Trả về |
|--|--|
| <code>get_n()</code> | 2 |
| <code>get_m()</code> | 1 |
| <code>ask(1,1)</code> | 1 |
| <code>guess(res) // với res=<2></code> | Kết thúc chương trình. Bạn đã trả lời đúng với số lượng câu hỏi là 1 và chương trình đạt điểm của ví dụ này. |

Hạn chế

- Subtask 1: có 30% số test ứng với $m = n - 1$; $a_1 < a_2 < \dots < a_m$.
- Subtask 2: có 30% số test ứng với $m = n - 2$;
- Subtask 3: 40% số test còn lại ứng với $0 \leq m \leq n$ và $m \neq n - 1$.

Bài B. Tô màu lưới

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 1 giây
Hạn chế bộ nhớ: 256 MB

Cho lưới ô vuông kích thước $m \times n$. Các hàng được đánh số từ 1 đến m , từ trên xuống dưới; các cột được đánh số từ 1 đến n , từ trái qua phải. Ô vuông thuộc hàng thứ i và cột thứ j có tọa độ (i, j) . Người ta tô các ô vuông bởi $2 \times n$ màu có mã màu được đánh số từ 1 đến $2 \times n$ sao cho mỗi màu đều được tô cho ít nhất một ô. Ký hiệu L_j là số lượng màu khác nhau được sử dụng để tô các ô trong cột thứ j ($j = 1, 2, \dots, n$). Ta gọi *độ đa sắc* của lưới là giá trị $\max_{1 \leq j \leq n} L_j$.

Cho phép thực hiện việc hoán đổi màu của hai ô ở hai đỉnh đối diện trên đường chéo của hình chữ nhật kích thước 2×3 bất kỳ. Mỗi phép hoán đổi được mô tả bởi bốn số nguyên (u, v, s, t) cho biết hai ô vuông (u, v) và (s, t) được hoán đổi màu.

Yêu cầu: Hãy xác định một dãy các phép hoán đổi màu để đưa lưới về trạng thái có độ đa sắc nhỏ nhất.

Dữ liệu vào

Vào từ thiết bị vào chuẩn: Dòng thứ nhất chứa số nguyên dương T ($T \leq 30$) là số lượng bộ dữ liệu. Mỗi nhóm dòng trong T nhóm dòng tiếp theo mô tả một bộ dữ liệu theo khuôn dạng sau:

- Dòng đầu tiên chứa 2 số nguyên m, n được ghi cách nhau bởi dấu cách;
- Dòng thứ i trong số m dòng tiếp theo chứa n số nguyên dương $c_{i1}, c_{i2}, \dots, c_{in}$ được ghi cách nhau bởi dấu cách, trong đó c_{ij} là mã màu của ô (i, j) trong lưới ban đầu ($j = 1, 2, \dots, n$).

Kết quả

Ghi ra thiết bị ra chuẩn T nhóm dòng, mỗi nhóm là kết quả tìm được cho bộ dữ liệu tương ứng trong dữ liệu vào, theo khuôn dạng sau:

- Dòng đầu tiên ghi ra số nguyên không âm p là số lượng phép hoán đổi cần thực hiện;
- Tiếp đến là p dòng mô tả dãy các phép hoán đổi cần thực hiện để đưa lưới về trạng thái có độ đa sắc nhỏ nhất. Mỗi dòng ghi 4 số nguyên dương u, v, s, t cách nhau bởi dấu cách cho biết cần thực hiện việc hoán đổi màu của hai ô vuông (u, v) và (s, t) .

Nếu có nhiều cách thực hiện để đưa lưới về trạng thái có độ đa sắc nhỏ nhất thì chỉ cần đưa ra một cách.

Ví dụ

| stdin | stdout |
|---------|---------|
| 2 | 0 |
| 4 4 | 2 |
| 1 2 3 4 | 2 2 4 3 |
| 5 6 7 8 | 2 2 4 1 |
| 1 2 3 4 | |
| 5 6 7 8 | |
| 4 4 | |
| 1 2 3 4 | |
| 5 7 7 8 | |
| 1 2 3 4 | |
| 6 6 5 8 | |

Hạn chế

- Subtask 1: $4 \leq m, n \leq 5$;
- Subtask 2: $6 \leq m, n \leq 50$.

Bài C. Chuỗi từ ghép

©thai9cdb

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 0.1 giây
Hạn chế bộ nhớ: 512 Mb

Cho W là một tập các từ, mỗi từ tạo bởi một hoặc nhiều kí tự viết liền nhau. Từ tập W , ta có thể tạo ra các chuỗi bằng cách viết liền các từ của W (mỗi từ có thể sử dụng nhiều lần). Ví dụ chuỗi: $S = x_1x_2 \dots x_k, x_i \in W, \forall i = \overline{1, k}$

Khi đó tập $X = \{x_1, x_2, \dots, x_k\}$ được gọi là một dẫn xuất của chuỗi S . Rõ ràng là một chuỗi có thể có nhiều dẫn xuất, ví dụ:

$W = \{ab, ba, a\}$
 $S = aba = (ab)a = a(ba)$

Yêu cầu: Hãy tìm một chuỗi có nhiều hơn một dẫn xuất. Hai dẫn xuất $X = \{x_1, x_2, \dots, x_k\}$ và $Y = \{y_1, y_2, \dots, y_k\}$ được coi là khác nhau nếu hoặc $k \neq q$ hoặc $\exists i \in \{1, \dots, k\} : x_i \neq y_i$.

Dữ liệu vào

- Dòng đầu tiên chứa một số nguyên dương n là lực lượng tập W ;
- Mỗi dòng trong số n dòng tiếp theo chứa một từ của W . Các từ chỉ chứa các chữ cái latin in thường, có độ dài nguyên dương và ≤ 1000 . Tổng độ dài các từ không quá 1000 và không có 2 từ nào giống nhau.

Kết quả

Nếu không tồn tại chuỗi nào có nhiều hơn 1 dẫn xuất, in ra -1. Ngược lại, in ra 2 dòng mô tả 2 dẫn xuất, mỗi dòng là chuỗi các từ của một dẫn xuất phân cách nhau bởi kí tự '+'. Xem thêm trong test ví dụ để rõ thêm về định dạng output. Nếu có nhiều hơn 2 dẫn xuất thì chỉ cần in ra 2 dẫn xuất bất kỳ.

Ví dụ

| stdin | stdout |
|--------------------|--------------|
| 3 ab ba a | a+ba ab+a |

Hạn chế

- Subtask 1: có 20% số test ứng với $n \leq 10$.
- Subtask 2: 80% số test còn lại ứng với giới hạn trong mô tả.