

TỔNG QUAN ĐỀ THI

Bài	Tên bài	Tên file chương trình	Hạn chế thời gian	Hạn chế bộ nhớ
1	r/place	PLACE.*	2 giây	1024MB
2	Biến đổi dãy số	TRANS.*	2 giây	1024MB
3	Chơi bài Yugi	YUGIOH.*	2 giây	1024MB
4	Đội cứu hộ	RESCUE.*	2 giây	1024MB

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

Bài 1. r/place

Dịp Cá tháng Tư năm nay, mạng xã hội Reddit tổ chức một sự kiện mang tên **r/place**, trong đó mỗi thành viên có thể tham gia tô màu trên một bảng canvas lớn để tạo nên một bức tranh rực rỡ, đặc sắc.

Bin vừa tham gia sự kiện này, vì hứng thú quá nên cậu quyết định rủ đám bạn tô bảng cùng mình. Do gia đình có điều kiện nên cậu sắm một bảng LED để tô nhiều màu chớp nhoáng.

Bảng đèn LED của Bin gồm  $M \times N$  bóng đèn, mỗi bóng đèn có thể sáng một trong  $K$  màu đã được lập trình trước, các màu này được thể hiện dưới dạng một số nguyên có giá trị từ 0 đến  $K - 1$ .

Bin và các bạn dự định sẽ thực hiện  $K$  thao tác trên bảng đèn này, các thao tác này thuộc một trong hai loại sau:

• *MOVE l r u d*:

- Thực hiện  $l$  lần thao tác: dịch các cột của bảng sang trái, riêng cột đầu tiên thì chuyển sang vị trí của cột cuối cùng
- Thực hiện  $r$  lần thao tác: dịch các cột của bảng sang phải, riêng cột cuối cùng thì chuyển sang vị trí của cột đầu tiên
- Thực hiện  $u$  lần thao tác: dịch các hàng của bảng lên trên, riêng hàng đầu tiên thì chuyển sang vị trí của hàng cuối cùng
- Thực hiện  $d$  lần thao tác: dịch các hàng của bảng xuống dưới, riêng hàng cuối cùng thì chuyển sang vị trí của hàng đầu tiên

- *QUERY*  $x_1 y_1 x_2 y_2$ : mỗi 1 giây, Bin có thể chọn một bóng đèn trên bảng và biến đổi màu của bóng đèn đó từ  $X$  sang  $X - 1$  hoặc  $X + 1$  (với điều kiện màu mới của pixel vẫn nằm trong đoạn  $[0, K - 1]$ ). Bin muốn biết cần ít nhất bao nhiêu giây để toàn bộ bóng đèn trong bảng con có tọa độ góc trái trên là  $(x_1, y_1)$  và góc phải dưới là  $(x_2, y_2)$  cùng màu?

Bin không muốn tốn sức giải quyết bài toán trên nên quyết định nhờ các bạn trả lời giúp bài toán này. Hãy giải quyết bài toán này một cách thật hiệu quả nhé!

**Dữ liệu:** Vào từ tập tin văn bản **PLACE.INP** :

- Dòng đầu tiên chứa 3 số nguyên dương  $N, M, K$ . ( $N, M \leq 500, K \leq 50$ ) là kích thước bảng đèn và số lượng màu có thể của mỗi bóng.

- $N$  dòng tiếp theo, mỗi dòng chứa  $M$  số nguyên không âm  $a_{i,j}$ . ( $a_{i,j} < k$ )

- Dòng tiếp theo gồm một số nguyên dương  $Q$  - số truy vấn ( $Q \leq 10^5$ ).

- $Q$  dòng tiếp theo, mỗi dòng thuộc 1 trong 2 loại: *MOVE*  $l r u d$  ( $1 \leq l, r, u, d \leq 10^9$ ) hoặc *QUERY*  $x_1 y_1 x_2 y_2$  ( $1 \leq x_1, y_1 \leq N, 1 \leq x_2, y_2 \leq M$ )

**Kết quả:** Ghi ra tập tin văn bản **PLACE.OUT** kết quả của các truy vấn *QUERY* trên các dòng riêng biệt

**Ví dụ:**

PLACE . INP	PLACE . OUT
3 3 5	4
3 4 2	3
0 3 1	
2 2 0	
3	
QUERY 1 1 2 2	
MOVE 2 0 0 1	
QUERY 1 1 2 2	

**Subtask:**

**Subtask 1:** (25%):  $N, M \leq 50, K \leq 20, Q \leq 10^3$ .

Với mỗi truy vấn *MOVE*:  $l + r + u + d \leq 20$ .

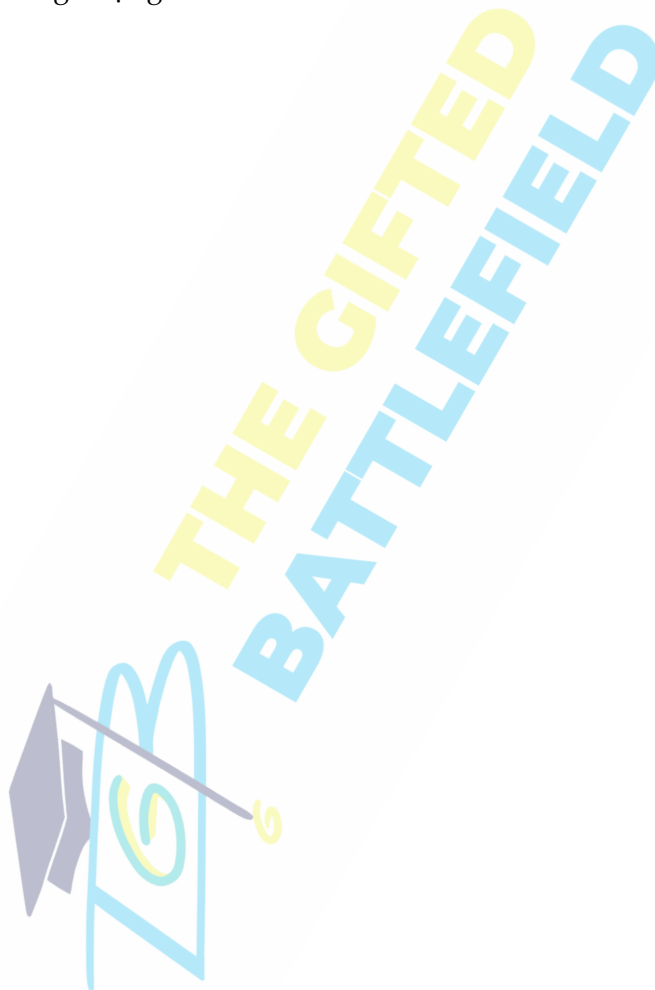
**Subtask 2:** (25%):  $K \leq 2$

Không có truy vấn *MOVE*.

**Subtask 2:** (25%):  $K \leq 50$

Không có truy vấn *MOVE*.

**Subtask 4:** (25%): không có ràng buộc gì thêm



## Bài 2. Biến đổi dãy số

Xét dãy số  $a$  gồm  $n$  phân tử. Định nghĩa các hàm sau:

- $p(x)$  = tích các chữ số của  $x$ .
- $s(x)$  = tổng các chữ số của  $x$ .
- $f(x) = p(x) + s(x)$ .

Trong mỗi bước, chọn một số  $a_i$  và biến  $a_i = f(a_i)$ .

Cho biết số bước biến đổi ít nhất để tạo ra một cặp số  $(i, j)$  ( $i \neq j$ ) sao cho  $a_i = a_j$ .

**Dữ liệu:** Vào từ tập tin văn bản **TRANS.INP**:

- Dòng đầu tiên chứa duy nhất 1 số nguyên  $n$  ( $n \leq 10^5$ ) là độ dài dãy  $a$ .
- Dòng thứ hai chứa  $n$  số nguyên dương  $a_i$  ( $a_i \leq 10^7$ ).

**Kết quả:** In ra một số nguyên duy nhất là số bước biến đổi để thỏa điều kiện đề bài **TRANS.OUT** một số nguyên duy nhất là số bước biến đổi để thỏa điều kiện đề bài. Nếu không có cách biến đổi thì in  $-1$ .

**Ví dụ:**

TRANS . INP	TRANS . OUT	GIẢI THÍCH
9 3 18 12 8 15 19 4 7 14	1	Biến đổi $a_7 = f(a_7) = f(4) = 8$ . Lúc này ta có $a_4 = a_7 = 8$

**Subtask:**

**Subtask 1:** (20%):  $n \leq 3, a_i \leq 10$

**Subtask 2:** (30%):  $n \leq 10^3, a_i \leq 10^5$

**Subtask 3:** (50%): không có ràng buộc gì thêm

### Bài 3. Chơi bài Yugi

MinhHo là một cậu bé ham chơi, đặc biệt cậu có niềm đam mê mãnh liệt với bộ môn trading card game, cụ thể là game Yugioh, một trò chơi thẻ bài rất thịnh hành hiện nay. Ở trò này, người chơi sẽ sử dụng bộ bài gồm những thẻ quái thú, ma pháp và chạp bẫy để làm tụt điểm gốc của đối phương từ 8000 về 0. Đặc biệt hơn, vào 1 lượt, một người chơi có thể kích hoạt nhiều thẻ cùng lúc để tạo nên một combo mạnh mẽ, một combo được kích hoạt khi chơi các thẻ có số thứ tự tăng dần và giá trị ghi ở trên 2 thẻ đó có chung ít nhất  $k$  ước nguyên tố, một combo càng mạnh khi nó gồm càng nhiều thẻ. Là một pro player đã luyện tập bộ bài "raidraptor" của mình đến mức nhuần nhuyễn, MinhHo luôn biết cách để chơi combo mạnh nhất gồm những thẻ trên tay mình và đặc biệt, một lượt cậu chỉ chơi 1 combo duy nhất, combo mạnh nhất mà cậu có thể ra. Tuy vậy, hôm nay cậu lại đang bị đau đầu do "múa" combo quá nhiều, bạn hãy giúp MinhHo ra combo trong lượt đầu này nhé!

**Dữ liệu:** Vào từ tập tin văn bản **YUGIOH.INP**:

- Dòng đầu tiên chứa 2 số nguyên  $n, k$  ( $n, k \leq 2 \cdot 10^3$ ) là số thẻ và điều kiện kích hoạt combo.
- Dòng thứ hai chứa  $n$  số nguyên dương  $a_i$  ( $a_i \leq 10^6$ ) là giá trị trên  $n$  thẻ.

**Kết quả:** Ghi ra tập tin văn bản **YUGIOH.OUT** số thẻ trong combo mạnh nhất mà MinhHo có thể ra.

**Ví dụ:**

YUGIOH.INP	YUGIOH.OUT	GIẢI THÍCH
5 2 6 24 10 20 12	3	Combo mạnh nhất gồm các thẻ bài 1, 2 và 5.

**Subtask:**

**Subtask 1:** (20%):  $n \leq 20$

**Subtask 2:** (30%):  $n \leq 100$

**Subtask 3:** (50%): không có ràng buộc gì thêm

#### Bài 4. Đội cứu hộ

Có  $k$  người đi tham quan mê cung. Mê cung có dạng ma trận 2 chiều  $m \times n$  và mỗi ô trong ma trận có chiều cao là  $h_{i,j}$  ( $1 \leq i \leq m, 1 \leq j \leq n$ ). Sau một hồi tham quan, mỗi người đều bí lối và bị kẹt lần lượt ở các ô  $(x_i, y_i)$  ( $1 \leq i \leq k$ ). Đội cứu hộ xuất phát tại vị trí  $(1, 1)$ . Họ sẽ di chuyển lần lượt trong mê cung để đến được vị trí của khách tham quan.

Biết rằng đội cứu hộ chỉ có thể di chuyển từ một ô sang các ô chung cạnh (không di chuyển ra khỏi mê cung) nếu chênh lệch chiều cao của 2 ô không vượt quá độ dài thang cứu hộ. Thang cứu hộ càng dài thì càng đắt tiền. Hãy giúp đội cứu hộ chọn chiếc thang ngắn nhất có thể mà vẫn cứu được  $k$  người tham quan ra khỏi mê cung. Đề đảm bảo vị trí  $k$  người là phân biệt.

**Dữ liệu:** Vào từ tập tin văn bản **RESCUE.INP**:

- Dòng đầu tiên chứa 2 số nguyên dương  $n, m$  ( $n, m \leq 10^3$ ) là kích thước của mê cung.
- $n$  dòng tiếp theo, mỗi dòng chứa  $m$  số nguyên dương  $h_{i,j}$  ( $h_{i,j} \leq 10^9$ ) là độ cao của các ô trên mê cung.
- Dòng tiếp theo chứa số nguyên dương  $k$  là số người bị mắc kẹt ( $1 \leq k \leq n \times m$ ).
- $k$  dòng tiếp theo, mỗi dòng chứa 2 số nguyên dương  $x_i, y_i$  là vị trí của người mắc kẹt thứ  $i$ .

**Kết quả:** Ghi ra tập tin văn bản **RESCUE.OUT** 1 số nguyên duy nhất - độ dài nhỏ nhất của thang cứu hộ để cứu được  $k$  người ra.

**Ví dụ:**

RESCUE.INP	RESCUE.OUT
3 4	2
1 3 2 5	
4 3 2 1	
3 4 1 2	
2	
1 2	
3 4	

**Subtask:**

**Subtask 1:** (30%):  $n \times m \leq 10^3, h \leq 10^3$

**Subtask 2:** (20%): Đáp án cần tìm  $\leq 100$

**Subtask 3:** (50%): không có ràng buộc gì thêm

===HẾT===

