

## Tổng quan bài thi ngày 1

Tên bài	BITFUN	FANS	MOVING
Tên chương trình	BITFUN.*	FANS.*	MOVING.*
Dữ liệu đầu vào	BITFUN.INP	FANS.INP	MOVING.INP
Dữ liệu đầu ra	BITFUN.OUT	FANS.OUT	MOVING.OUT
Giới hạn thời gian	1s	1s	1s
Giới hạn bộ nhớ	1024MB	1024MB	1024MB
Điểm	7	7	6

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

## Thông tin bổ sung

- Thời lượng: 180 phút.
- Thời gian: 8:00 - 11:00 hoặc 14:00 - 17:00 ngày 22/12/2018.
- Nén bài làm dưới dạng PreVOI\_x.zip", trong đó x là số thứ tự được cấp trong file **Danh sách PreVOI Đà Lạt 2018** (nhấn vào đường dẫn).
- Để đặt câu hỏi về đề bài, truy cập vào [sli.do](https://sli.do), event code là N703.
- Lưu ý khi làm bài, với những bài yêu cầu nhập xuất nhiều dữ liệu, **ên sử dụng Fast I/O thay cho nhập xuất bình thường** nhằm tránh bị TLE đáng tiếc.

## Môi trường

- Cấu hình máy chấm: intel core i7-6500u cpu @ 2.50ghz
  - Môi trường chấm bài: Themis
  - Câu lệnh biên dịch:  
C : gcc -O2 -std=gnu11 -Wl,-stack,1073741824 \$files  
C++ : g++ -O2 -std=c++11 -Wl,-stack,1073741824 \$files  
Pascal : fpc -O2 \$files
-

## BITFUN

AFF Cup 2018 - Chuyện giờ mới kể:

Trước thềm trận chung kết lượt về Việt Nam vs Malaysia, nhằm giải tỏa căng thẳng chuẩn bị nhận cúp Vô Địch, Xuân Trường và Đức Chinh nghĩ ra một trò chơi gồm  $T$  vòng đấu.

Luật chơi như sau :

Để bắt đầu một vòng đấu, họ ngẫu nhiên chọn ra một số nguyên  $N$ .

- Hai cầu thủ sẽ lần lượt chơi. Xuân Trường là người đi trước.
- Trong mỗi lượt đi, người chơi đổi vị trí 2 bit bất kì trong biểu diễn nhị phân của số  $N$  để tạo thành số  $N$  mới, sao cho số sau khi đổi chỗ phải NHỎ HƠN số trước đó.
- Người nào không thể thực hiện phép đổi chỗ khi đến lượt mình sẽ thua cuộc.

Trước khi trở thành cầu thủ xuất sắc, họ rất giỏi môn Tin học và cực kì am hiểu trò chơi với các con số (chẳng qua họ không biết điều đó thôi), cả hai đều muốn giành chiến thắng như cách họ sắp vô địch AFF Cup vậy. Bạn hãy giúp Xuân Trường xác định xem anh ấy có thể thắng trò chơi nếu cả hai cùng chơi theo cách tối ưu nhất không nhé!

## Dữ liệu

- Dòng đầu tiên chứa số nguyên dương  $T$  ( $1 \leq T \leq 10^5$ ) - là số lượng vòng đấu.
- $T$  dòng tiếp theo mỗi dòng chứa một số nguyên  $N$ .

## Kết quả

- Gồm  $T$  dòng ứng với  $T$  vòng đấu, nếu Xuân Trường có thể thắng trong vòng đấu đó hãy in ra "YES", ngược lại in "NO" (không có dấu " ").

## Ví dụ

BITFUN.INP	BITFUN.OUT
3	NO
3	YES
5	YES
10	

## Giới hạn

- Subtask 1 (10%):  $N = 2^k$  hoặc  $N = 2^k - 1$  với  $0 \leq k \leq 30$ .
- Subtask 2 (20%):  $0 \leq N \leq 10^3$ .
- Subtask 3 (40%):  $0 \leq N \leq 10^5$ .
- Subtask 4 (30%):  $0 \leq N \leq 2^{30}$  và số lượng bit 1 trong biểu diễn nhị phân của  $N$  không vượt quá 4.

## Giải thích

Trong test ví dụ đầu tiên:

- Ở vòng đấu thứ nhất, biểu diễn  $N = 3$  dưới dạng một dãy nhị phân là 011, không tồn tại phép biến đổi nào để thu được số nhỏ hơn nên vì vậy Xuân Trường không thể thắng cuộc.
- Vòng đấu thứ hai, biểu diễn  $N = 5$  dưới dạng một dãy nhị phân là 101. Trong lượt của mình, Xuân Trường biến đổi thành dãy nhị phân 011. Vì lượt sau Đức Chinh không thể thực hiện phép biến đổi để số thu được nhỏ hơn số trước đó nên Xuân Trường thắng.

## FANS

AFF Cup 2018 - Cổ động viên (CDV):

AFF Cup không chỉ đơn thuần là một giải vô địch bóng đá Đông Nam Á, mà còn là một giải thể thao giúp thúc đẩy sự hợp tác, sự hiểu biết và sự gắn kết các mối quan hệ giữa các nước khu vực Đông Nam Á. Nhận thấy điều này, ban quản lý sân vận động quốc gia Mỹ Đình quyết định xếp các cổ động viên Việt Nam và Malaysia ngồi chung với nhau. Một dãy ghế có đúng  $N$  ghế ngồi. Để tránh trường hợp các CDV quá khích dẫn đến mất tinh thần đoàn kết hai nước, phải đảm bảo tính cân bằng : ở mọi đoạn ghế liên tiếp, số lượng CDV hai nước không được chênh lệch quá 2. Một dãy ghế sẽ được biểu diễn dưới dạng xâu gồm hai kí tự  $V$  - CDV Việt Nam và  $M$  - CDV Malaysia.

Ví dụ, với  $N = 5$  có tất cả 14 dãy ghế đảm bảo tính cân bằng. Theo thứ tự từ điển, các dãy ghế đó sẽ lần lượt là: MMVMV, MMVVM, MVMMV, MVMVM, MVMVV, MVVMM, MVVMV, VMMVM, VMMVV, VMVMM, VMVMV, VMVVM, VVMMV. Các dãy ghế được đánh số từ 1 trở đi. Ví dụ, với  $N = 5$ , dãy ghế thứ 5 là "MVMVV".

Ban quản lý có  $T$  câu hỏi cho bạn. Mỗi câu hỏi cho biết số ghế  $N$  của dãy ghế và xâu biểu diễn dãy ghế cân bằng, yêu cầu bạn hãy tính số thứ tự của dãy ghế này theo modulo  $M$  cho trước.

Nếu modulo  $M = 0$  thì in ra thứ tự của dãy ghế không theo modulo. Nói cách khác, với  $M = 0$  thì in ra giá trị thực của kết quả.

## Dữ liệu

- Dòng đầu tiên chứa hai số nguyên dương  $T$  ( $1 \leq T \leq 20$ ) là số lượng yêu cầu..
- $2 \times T$  dòng sau:
- Dòng thứ  $2 \times i$  chứa hai số nguyên  $N, M$ .
- Dòng thứ  $2 \times i + 1$  chứa xâu gồm  $N$  kí tự  $V$  hoặc  $M$  biểu diễn dãy ghế.

## Kết quả

- In ra  $T$  dòng là kết quả ứng với  $T$  yêu cầu.

## Ví dụ

FANS . INP	FANS . OUT
3	5
5 10	1
MVMVV	39
5 4	
MVMVV	
12 10000	
MVMMVMVVMVMM	

## Giới hạn

- Subtask 1 (20%):  $1 \leq N \leq 18$  và  $1 \leq M \leq 10^9$ .
- Subtask 2 (40%):  $1 \leq N \leq 20000$  và  $1 \leq M \leq 10^9$ .
- Subtask 3 (40%):  $1 \leq N \leq 200$  và  $M = 0$ .

## MOVING

AFF Cup nào đó - Nhập vai:

Sau những ngày tháng lập trình miệt mài, vất vả, cuối cùng bạn cũng trở thành phụ tá huấn luyện viên cho đội bóng tham gia giải đấu AFF Cup 20xx. Đội hình các cầu thủ trên sân được biểu diễn bằng một ma trận hai chiều có kích thước  $N \times M$  (ma trận có  $N$  dòng và  $M$  cột). Mỗi ô  $(i, j)$  (là ô giao giữa hàng thứ  $i$  và cột thứ  $j$ ,  $i$  và  $j$  được đánh số từ 1) chứa một số nguyên 0 hoặc 1, 1 nếu có một cầu thủ nào đó đang đứng tại ô  $(i, j)$  và 0 nếu ô đó trống.

Bạn được điều khiển đội bóng trong  $T$  trận đấu.

Mỗi trận đấu bạn cần điều chỉnh đội hình trên sân hiện tại (được biểu diễn bằng ma trận  $A$ ) thành đội hình mong muốn (được biểu diễn bằng ma trận  $B$ ). Tất nhiên số lượng cầu thủ trong đội hình trên sân và trong đội hình mong muốn là bằng nhau.

Huấn luyện viên trưởng cho  $C$  là một ma trận hai chiều kích thước  $N \times M$ , mỗi ô  $(i, j)$  chứa một số nguyên biểu diễn số lần bạn được phép điều khiển cầu thủ của mình đến ô  $(i, j)$ . Tại mỗi thời điểm, chỉ một cầu thủ trên sân được phép di chuyển đến 4 ô kề cạnh, tuy nhiên **KHÔNG ĐƯỢC PHÉP**:

- Đi ra khỏi sân.
- Đi đến ô  $(i, j)$  khi ô đó đã được đi  $C(i, j)$  lần.
- Đi vào ô đã có người đứng. Nói cách khác, tại mỗi thời điểm, không được có 2 cầu thủ đứng cùng một ô.

Bạn đáng ra đã có thể thoải mái điều chỉnh đội hình. Nhưng vì cộng thêm yêu cầu "khó đỡ" từ vị huấn luyện viên trưởng nên có thể bạn không thể tạo ra đội hình như mong muốn. Tuy nhiên, bạn cần tìm ra số lượng lớn nhất các cầu thủ có thể đứng ở những vị trí có cầu thủ trong đội hình mong muốn (giả sử vai trò của các cầu thủ là như nhau). Nói cách khác, sau khi biến đổi bảng  $A$ , ta định nghĩa thêm mảng  $S$  có kích thước  $N \times M$ ,  $S(i, j) = 1$  nếu  $A(i, j) = B(i, j) = 1$  và  $S(i, j) = 0$  trong những trường hợp còn lại. Mục tiêu đặt ra là tìm cách biến đổi mảng  $A$  sao cho tổng giá trị các phần tử của mảng  $S$  là lớn nhất. In ra giá trị đó.

## Dữ liệu

- Dòng đầu tiên chứa số nguyên dương  $T$  ( $1 \leq T \leq 20$ ) là số lượng trận đấu.
  - $T$  trận đấu có dạng :
    - Dòng đầu tiên chứa 2 số nguyên  $N, M$  là kích thước ma trận  $A, B, C$ .
    - $N$  dòng sau đó, mỗi dòng chứa  $M$  số nguyên 0 hoặc 1 mô tả ma trận  $A$ .
    - $N$  dòng sau đó, mỗi dòng chứa  $M$  số nguyên 0 hoặc 1 mô tả ma trận  $B$ .
    - $N$  dòng sau đó, mỗi dòng chứa  $M$  số nguyên mô tả ma trận  $C$  (mọi giá trị của ma trận  $C$  đều nhỏ hơn 1000).
-

Mỗi số cách nhau ít nhất một khoảng trắng.

## Kết quả

- In ra  $T$  dòng là kết quả ứng với  $T$  trận đấu.

## Ví dụ

MOVING.INP	MOVING.OUT
2	3
4 4	1
1 1 0 0	
0 1 0 0	
0 0 0 0	
1 0 0 0	
1 0 0 1	
0 0 0 0	
0 0 0 1	
0 1 0 0	
0 0 0 4	
0 1 1 1	
0 1 1 1	
5 0 0 0	
1 5	
1 1 0 0 0	
0 0 0 1 1	
10 10 1 10 10	

## Giới hạn

- Subtask 1 (5%):  $1 \leq N, M \leq 30$ ,  $C(i,j)$  được sinh bằng công thức:  
 $C(i,j) = \text{rand}() \bmod 101 + 900$ , trong đó  $\text{rand}()$  là hàm lấy số ngẫu nhiên,  $\bmod$  là phép chia lấy dư.
- Subtask 2 (5%):  $1 \leq N, M \leq 30$  và trên sân có đúng 1 cầu thủ.
- Subtask 3 (10%):  $1 \leq N, M \leq 30$  và trên sân có đúng 2 cầu thủ.
- Subtask 4 (20%):  $1 \leq N, M \leq 30$  và trên sân có đúng 3 cầu thủ.
- Subtask 5 (60%):  $1 \leq N, M \leq 30$  và không có ràng buộc gì thêm.

## Giải thích

- Trong test ví dụ đầu tiên. Một cách di chuyển tối ưu là: Giữ nguyên vị trí cầu thủ đứng ở ô (1, 1), di chuyển cầu thủ ở ô (2, 2) theo thứ tự (2, 2) - (3, 2) - (3, 3) - (3, 4), di chuyển cầu thủ ở ô (1, 2) theo thứ tự (1, 2) - (2, 2) - (2, 3) - (2, 4) - (1, 4), cầu thủ ở ô (4, 1) không thể di chuyển. Có 3 cầu thủ ở vị trí đội hình mong muốn là (1, 1), (3, 4) và (1, 4). Tuy ô (2, 2) xuất hiện 2 lần và  $C(2,2) = 1$  nhưng lần đầu tiên không được tính vì đó là vị trí xuất phát của đường đi.