

**Bài 1. Thi đấu**

BATTLE.[CPP|INP|OUT]

Nhóm A có  $N$  thí sinh, mỗi thí sinh có sức mạnh bằng  $a[i]$ , nhóm B có  $M$  thí sinh, mỗi thí sinh có sức mạnh bằng  $b[i]$ .

Luật thi đấu đối kháng như sau: Mỗi nhóm chọn ra  $K$  thí sinh, thí sinh mạnh nhất được chọn của nhóm A sẽ thi đấu với thí sinh mạnh nhất của nhóm B, thí sinh mạnh thứ 2 của nhóm A sẽ thi đấu với thí sinh mạnh thứ 2 trong nhóm B... Trong một cuộc đấu đối kháng, thí sinh nào có sức mạnh lớn hơn sẽ chiến thắng.

Bạn muốn tính xem có bao nhiêu cách lựa chọn  $K$  thí sinh nhóm A và  $K$  thí sinh nhóm B sao cho trong  $K$  cuộc đấu, thí sinh đến từ nhóm A luôn chiến thắng.

**Dữ liệu:** BATTLE.INP

- Dòng đầu tiên chứa 3 số nguyên  $N, M, K$  ( $1 \leq N, M \leq 1000, 1 \leq K \leq 10$ ).
- Dòng tiếp theo gồm  $N$  số nguyên  $a[i]$ .
- Dòng cuối gồm  $M$  số nguyên  $b[i]$  ( $1 \leq a[i], b[i] \leq 10^9$ ).

**Kết quả:** BATTLE.OUT

- In ra đáp án tìm được theo modulo  $10^9+9$

BATTLE.INP	BATTLE.OUT
5 10 3	2
1 2 2 6 7	
1 3 6 8 8 9 14 17 18 19	

Giải thích test: Nhóm A chọn (2, 6, 7), nhóm B chọn (1, 3, 6). Nhóm A có 2 cách chọn bộ (2, 6, 7) tương ứng với 2 cách.

**Giới hạn:** 50% số test có  $N$  hoặc  $M \leq 10$

**Bài 2. LƯU TRỮ**

STORAGE.[CPP|INP|OUT]

Có  $n$  đồ vật đánh số từ 1 đến  $n$  nằm rải rác trên sàn và có  $k$  thùng đánh số từ 1 đến  $k$ . Bạn quyết định dọn dẹp, bỏ đồ vào trong thùng, mỗi đồ vật sẽ được bỏ vào một thùng. Để tiện cho việc tìm kiếm sau này, Bạn quyết định bỏ đồ vật thứ  $i$  vào một trong 2 thùng  $a_i$  hoặc  $b_i$ .

Bạn nhặt lần lượt các đồ vật từ 1 đến  $n$  và cất đồ vật thứ  $i$  theo quy tắc đầu tiên có thể chọn trong số các quy tắc sau:

- Nếu thùng  $a_i$  rỗng thì cất vào thùng này,
- Nếu thùng  $b_i$  rỗng thì cất vào thùng này,

- Cố gắng chuyển đồ từ thùng  $a_i$  sang thùng khác phù hợp theo quy định và cứ di chuyển tiếp cho đến khi giải phóng được thùng  $a_i$  để lưu trữ, nếu không giải phóng được thì áp dụng quy tắc tiếp theo,
- Cố gắng chuyển đồ từ thùng  $b_i$  sang thùng khác phù hợp theo quy định và cứ di chuyển tiếp cho đến khi giải phóng được thùng  $b_i$  để lưu trữ, nếu không giải phóng được thì áp dụng quy tắc tiếp theo,
- Vứt đồ vật này.

Hãy xác định những đồ vật nào lưu trữ được và đồ vật nào phải vứt bỏ. Với đồ vật lưu trữ được ghi ra số 1, với đồ vật phải vứt bỏ ghi ra số 0.

**Dữ liệu:** Vào từ file văn bản STORAGE.INP:

- Dòng đầu tiên chứa 2 số nguyên  $n$  và  $k$  ( $1 \leq n, k \leq 3 \times 10^5$ ),
- Dòng thứ  $i$  trong  $n$  dòng sau chứa 2 số nguyên  $a_i$  và  $b_i$  ( $1 \leq a_i, b_i \leq k, a_i \neq b_i$ ).

**Kết quả:** Đưa ra file văn bản STORAGE.OUT xâu lần lượt ghi trạng thái đồ vật được lưu trữ hay vứt bỏ xác định được.

**Ví dụ:**

STORAGE.INP	STORAGE.OUT
9 10	111111111
1 2	
3 4	
5 6	
7 8	
9 10	
2 3	
1 5	
8 2	
7 9	

### **Bài 3. Lâu đài**

CASTLE.[CPP|INP|OUT]

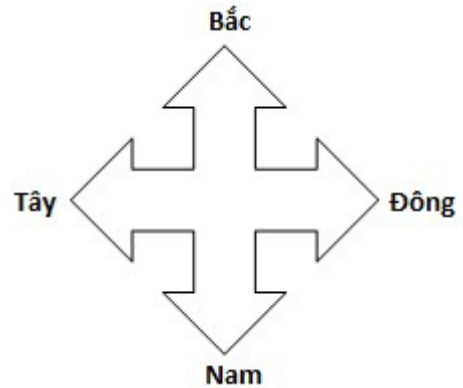
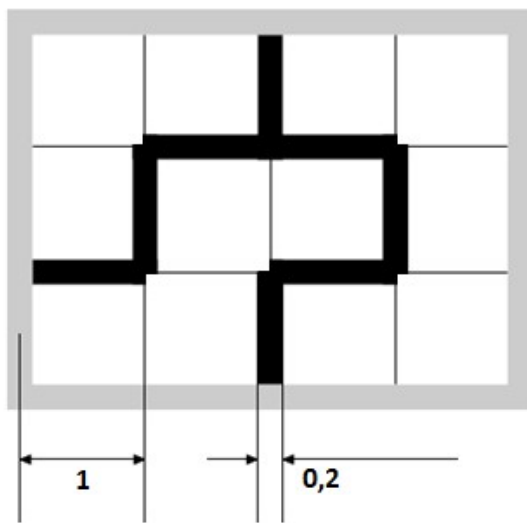
Một toà lâu đài hình chữ nhật kích thước  $m \times n$  được chia thành lưới ô vuông đơn vị. Các dòng từ trên xuống được đánh số từ 1 tới  $m$ , các cột từ trái qua phải được đánh số từ 1 tới  $n$ . Quanh mỗi ô có thể có từ 0 tới 4 bức tường. Mỗi bức tường có chiều dài là 1 đơn vị và chiều rộng là 0,2 đơn vị. Quanh lâu đài có tường bao bọc. Để thể hiện tình trạng tường quanh một ô, ta gán cho mỗi ô một số nguyên, mà trong biểu diễn nhị phân của số nguyên đó:

- Bit 0 bằng 1 hay 0 tùy theo ô đó có tường hay không có tường hướng Tây
- Bit 1 bằng 1 hay 0 tùy theo ô đó có tường hay không có tường hướng Bắc
- Bit 2 bằng 1 hay 0 tùy theo ô đó có tường hay không có tường hướng Đông
- Bit 3 bằng 1 hay 0 tùy theo ô đó có tường hay không có tường hướng Nam

Ví dụ trong hình vẽ dưới, ta có một lâu đài  $3 \times 4$ .

Tình trạng tường của ô (2, 2) được thể hiện bởi số  $3_{10} = 0011_2$

Tình trạng tường của ô (1,3) được thể hiện bởi số  $11_{10} = 1011_2$



Lâu đài được chia thành các phòng, các phòng phân cách nhau bởi các bức tường. Hãy lập chương trình trả lời các câu hỏi sau:

1. Cho biết lâu đài có bao nhiêu phòng?
2. Cho biết diện tích của phòng rộng nhất?
3. Nếu được phá đi một và chỉ một bức tường thì được phòng có diện tích lớn nhất là bao nhiêu?

**Dữ liệu:** Vào từ file văn bản CASTLE.INP

- Dòng 1: Ghi hai số  $m, n$  ( $1 \leq m, n \leq 100$ )

-  $m$  dòng tiếp theo, dòng thứ  $i$  ghi  $n$  số nguyên, số thứ  $j$  thể hiện tình trạng tường quanh ô  $(i, j)$

Dữ liệu vào bảo đảm tình trạng có tường tại các ô kề cạnh là không mâu thuẫn nhau và có ít nhất 2 phòng, các số trên một dòng được ghi cách nhau ít nhất một dấu cách.

**Kết quả:** Ghi ra file văn bản CASTLE.OUT

- Dòng 1: Ghi số phòng;

- Dòng 2: Ghi diện tích của phòng rộng nhất;

- Dòng 3: Ghi diện tích của phòng rộng nhất thu được sau khi phá tường.

(Kết quả câu 2 và câu 3 ghi theo quy cách gồm 2 chữ số sau dấu chấm thập phân)

CASTLE.INP	CASTLE.OUT
2 1	2
15	0.64
3 4	3
3 14 11 6	3.86
13 3 14 5	7.10