

## 1. DU LỊCH NHIỀU TUA NHẤT

Một khu thắng cảnh gồm  $n$  điểm đánh số từ 1 tới  $n$  và  $m$  đường đi hai chiều. Mỗi đường đi nối hai địa điểm khác nhau trong số  $n$  địa điểm đã cho và giữa hai địa điểm bất kỳ có nhiều nhất một đường đi nối chúng. Một Tour du lịch là một hành trình xuất phát từ một địa điểm, đi thăm ít nhất hai địa điểm khác và quay trở về điểm xuất phát. Ngoại trừ địa điểm xuất phát, không địa điểm nào bị thăm tới hai lần.

*Yêu cầu: Hãy tìm một số tour du lịch nhiều nhất sao cho mỗi tour du lịch tìm được đều có một đoạn đường riêng hoàn toàn không có mặt trong các tua du lịch còn lại.*

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

- Dòng 1: Ghi hai số  $n, m$  ( $1 \leq n, m \leq 20000$ )
- $m$  dòng tiếp theo mỗi dòng ghi hai số nguyên dương  $x, y$  tương ứng với một đường đi trực tiếp nối địa điểm  $x$  với địa điểm  $y$

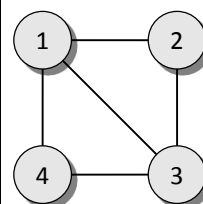
**Kết quả:** Ghi ra file văn bản TOURS.OUT

- Dòng 1: Ghi số  $k$  là số tour du lịch tìm được
- $k$  dòng tiếp theo, dòng thứ  $i$  mô tả tour du lịch thứ  $i$ : bắt đầu là địa điểm xuất phát, tiếp theo là danh sách các địa điểm sẽ đi tiếp theo thứ tự trong hành trình, kết thúc là địa điểm xuất phát.

*Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách*

**Ví dụ:**

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



## 2. TẬP KẾT

Một bàn cờ kích thước  $m \times n$  trong đó đánh dấu một số ô cấm. Các hàng của bàn cờ được đánh số từ 1 tới  $m$  theo thứ tự từ trên xuống dưới và các cột của bàn cờ được đánh số từ 1 tới  $n$  theo thứ tự từ trái qua phải. Trên bàn cờ có  $k$  quân mã đang đứng ở những vị trí khác nhau. Cần đi những quân mã này đến  $k$  vị trí tập kết khác nhau, mỗi quân mã một vị trí. Trong quá trình di chuyển, mỗi bước mã phải nhảy theo một nước đi của quân mã theo luật cờ, không được nhảy đến các ô cấm và không được nhảy đến ô đang có quân mã khác đang đứng. Vai trò của các quân mã và các vị trí tập kết là như nhau (một quân mã có thể cho đi tới bất kỳ vị trí tập kết nào nếu có đường nhảy).

**Yêu cầu:** Lập chương trình xác định cách đi các quân mã sao cho tổng số bước đi của các quân mã là nhỏ nhất.

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

- Dòng 1: Ghi hai số nguyên dương  $m, n$
- $m$  dòng tiếp theo, dòng  $i$  ghi  $n$  ký tự thể hiện hàng  $i$  của bàn cờ. Ký tự thứ  $i$  có thể là:
  - “.”: Thể hiện ô trống
  - “#”: Thể hiện ô cấm
  - “\$”: Thể hiện ô có mã đang đứng
  - “@”: Thể hiện ô ở vị trí tập kết

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

- Dòng 1 ghi tổng số bước di chuyển để đưa các quân mã đến vị trí tập kết ( $p$ )
- $p$  dòng tiếp theo liệt kê các bước di chuyển theo thứ tự thực hiện: mỗi dòng ghi bốn số nguyên  $x_1, y_1, x_2, y_2$  cách nhau đúng một dấu cách, cho biết bước di chuyển tương ứng sẽ chuyển quân mã ở ô  $(x_1, y_1)$  tới ô  $(x_2, y_2)$

**Ràng buộc:**

$m, n, k \leq 100$ ; Dữ liệu vào luôn đảm bảo có phương án thực hiện yêu cầu đề ra. Các số trên một dòng của input/output files được/phải ghi cách nhau ít nhất một dấu cách

**Ví dụ:**

KNIGHTS.INP	KNIGHTS.OUT
6 6	7
#.#..\$	5 4 4 6
...\$@.	4 6 2 5
..#...	4 3 5 5
..\$#..	1 6 3 5
##. \$@@	3 5 5 6
#....@	2 4 4 5
	4 5 6 6

### 3. DÃY SỐ

Cho một dãy gồm  $n$  số nguyên  $A = (a_1, a_2, \dots, a_n)$  và một số nguyên  $k$ . Hãy xác định xem trong dãy  $A$  có tồn tại hai phần tử  $a_p, a_q$  ở hai vị trí khác nhau  $p \neq q$  mà  $a_p - a_q = k$  hay không.

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

- Dòng 1: Chứa hai số nguyên  $n$  và  $k$  ( $2 \leq n \leq 10^5, |k| \leq 2 \cdot 10^9$ )
- Dòng 2: Chứa  $n$  số nguyên  $a_1, a_2, \dots, a_n$  ( $\forall i: |a_i| \leq 2 \cdot 10^9$ )

**Kết quả:** Ghi ra file văn bản SEQ.OUT hai chỉ số  $p, q$  tìm được. Nếu không tồn tại cặp số thỏa mãn yêu cầu, ghi ra hai số 0

*Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách*

**Ví dụ:**

SEQ. INP	SEQ. OUT
7 88	7 1
11 33 55 99 33 77 99	

## 4. XỔ SỐ

Công ty điện thoại LuckyPhone tổ chức xổ số để quyên góp tiền cho quỹ hỗ trợ tài năng trẻ nước NumberLand. Luật chơi như sau:

- Có  $n$  khách hàng tham gia chơi ( $1 \leq n \leq 10^6$ ) mỗi khách hàng tự chọn một số tự nhiên có không quá 9 chữ số, sau đó soạn tin nhắn gửi số mình chọn đến số điện thoại 19001234. Mỗi khách hàng chỉ được gửi đúng một tin nhắn.
- Trong các số nhận được từ các khách hàng, số lớn nhất trong các số được ít khách hàng gửi đến nhất sẽ được chọn làm số may mắn (Lucky Number).
- Công ty thu từ mỗi khách hàng đã gửi tin nhắn  $x$  đồng ( $1 \leq x \leq 10^9$ ) và phải tặng cho tất cả các khách hàng đã gửi số may mắn mỗi người một giải thưởng giá trị đúng bằng số may mắn.
- Lợi nhuận của công ty thu được trong cuộc chơi bằng tổng số tiền thu được từ các khách hàng trừ đi số tiền trao giải thưởng.

*Yêu cầu: Hãy tính lợi nhuận của công ty LuckyPhone trong cuộc chơi*

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

- Dòng 1: Chứa hai số nguyên dương  $n, x$
- $n$  dòng tiếp theo, mỗi dòng chứa một số nhận được từ một khách hàng

**Kết quả:** Ghi ra file văn bản LOTTERY.OUT lợi nhuận thu được

*Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách*

**Ví dụ:**

LOTTERY.INP	LOTTERY.OUT
9 100	12
2	
33	
1	
2	
444	
33	
1	
1	
444	

Giải thích: Trong ví dụ trên, có 3 số ít được chọn nhất là 2, 33 và 444 (mỗi số được chọn 2 lần). Theo quy tắc đặt ra, số may mắn là số 444. Tổng tiền thu được từ khách hàng là  $9 \times 100 = 900$ , tổng tiền trả thưởng là  $2 \times 444 = 888$ . Vậy lợi nhuận của công ty trong cuộc chơi là  $900 - 888 = 12$

## 5. LẮP RÁP Ô TÔ

Để tự động hoá dây chuyền lắp ráp ô tô, người ta sử dụng một robot lắp ráp và  $n$  dụng cụ lắp ráp đánh số từ 1 tới  $n$ . Có tất cả  $m$  loại bộ phận trong một chiếc ô tô đánh số từ 1 tới  $m$ . Mỗi chiếc ô tô phải được lắp ráp từ  $t$  bộ phận  $O = (o_1, o_2, \dots, o_t)$  theo đúng thứ tự này ( $1 \leq o_i \leq m, \forall i$ ). Biết được những thông tin sau:

- Tại mỗi thời điểm, robot chỉ có thể cầm được 1 dụng cụ.
- Tại thời điểm bắt đầu, robot không cầm dụng cụ gì cả và phải chọn một trong số  $n$  dụng cụ đã cho, thời gian chọn không đáng kể.
- Khi đã có dụng cụ, robot sẽ sử dụng nó để lắp một bộ phận trong dãy  $O$ , biết thời gian để Robot lắp bộ phận loại  $v$  bằng dụng cụ thứ  $i$  là  $b_{iv}$  ( $1 \leq i \leq n; 1 \leq v \leq m$ )
- Sau khi lắp xong mỗi bộ phận, robot được phép đổi dụng cụ khác để lắp bộ phận tiếp theo, biết thời gian đổi từ dụng cụ  $i$  sang dụng cụ  $j$  là  $a_{ij}$ . (Lưu ý rằng  $a_{ij}$  có thể khác  $a_{ji}$  và  $a_{ii}$  luôn bằng 0).

**Yêu cầu:** Hãy lập trình cho robot lắp ráp chiếc ô tô từ các bộ phận  $O = (o_1, o_2, \dots, o_t)$  trong thời gian ngắn nhất.

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

- Dòng 1: Chứa 3 số nguyên dương  $n, m, t \leq 200$
- Dòng 2: Chứa  $t$  số nguyên dương  $o_1, o_2, \dots, o_t$
- $n$  dòng tiếp theo, dòng thứ  $i$  chứa  $n$  số nguyên  $a_{i1}, a_{i2}, \dots, a_{in} \leq 200$
- $n$  dòng tiếp theo, dòng thứ  $j$  chứa  $m$  số nguyên  $b_{j1}, b_{j2}, \dots, b_{jm} \leq 200$

**Kết quả:** Ghi ra file văn bản VITERBI.OUT một số nguyên duy nhất là thời gian ít nhất để lắp ráp xong toàn bộ  $t$  linh kiện  $O = (o_1, o_2, \dots, o_t)$ .

Các số trên một dòng của Input files được ghi cách nhau ít nhất một dấu cách

**Ví dụ:**

VITERBI . INP	VITERBI . OUT
3 4 8	21
1 2 3 4 1 2 3 4	3 2 1 1 3 2 1 1
0 9 1	
1 0 9	
9 1 0	
8 8 1 5	
8 1 8 8	
1 8 8 5	

## 6. LÁT GẠCH

Phú ông thách Bờm chơi trò chơi lát gạch. Trò chơi diễn ra trên sân nhà Phú ông có bản đồ là một hình chữ nhật kích thước  $2m \times 2n$  ô vuông đơn vị. Trên bản đồ của sân chơi,  $2m$  hàng ô vuông được đánh số từ 1 tới  $2m$  theo thứ tự từ trên xuống dưới và  $2n$  cột ô vuông được đánh số từ 1 tới  $2n$  theo thứ tự từ trái qua phải. Trên mỗi ô vuông của sân ghi một số nguyên.

Bờm phải lát kín sân bằng các viên gạch kích thước  $1 \times 2$  hoặc  $2 \times 1$  sao cho mỗi viên gạch lát khớp đúng hai ô kề cạnh nhau của sân và mỗi ô chỉ có đúng một viên gạch lát lên nó. Khi lát mỗi viên gạch, gọi  $p$  là tích của hai số ghi trên hai ô được viên gạch đó lát lên. Nếu tích này là số dương, Bờm sẽ lấy của Phú ông một số tiền bằng  $p$ , nếu tích này là số âm thì Bờm mất cho Phú ông một số tiền bằng  $-p$ . Khi kết thúc trò chơi, ký hiệu tổng số tiền Bờm lấy được của Phú ông là  $\alpha$  và tổng số tiền Bờm mất cho Phú ông là  $\beta$ .

**Hãy giúp Bờm lát gạch sao cho khi kết thúc trò chơi, hiệu số  $\alpha - \beta$  là lớn nhất có thể.**

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

- Dòng 1: Chứa hai số nguyên dương  $m, n \leq 100$
- $2m$  dòng tiếp theo, dòng thứ  $i$  chứa  $2n$  số nguyên có giá trị tuyệt đối không quá 1000, số thứ  $j$  là số ghi trên ô  $(i, j)$  của sân

**Kết quả:** Ghi ra file văn bản BRICKS.OUT một số nguyên duy nhất là hiệu số  $\alpha - \beta$  tìm được

Các số trên một dòng của Input file được ghi cách nhau ít nhất một dấu cách

**Ví dụ:**

BRICKS.INP	BRICKS.OUT
1 2 1 2 5 6 3 4 7 -8	1

1	2	5	6
3	4	7	-8

## 7. HỘI CHỢ

Bản đồ hội chợ là một hình chữ nhật được chia thành lưới ô vuông đơn vị với các hàng đánh số từ 1 tới  $m$  theo thứ tự từ trên xuống dưới và các cột được đánh số từ 1 tới  $n$  theo thứ tự từ trái qua phải. Mỗi ô tượng trưng cho một gian hàng. Đến thăm gian hàng ở ô  $(i, j)$  thì phải trả một số tiền là  $a_{ij}$ . Quy ước rằng nếu  $a_{ij} = 0$  thì  $(i, j)$  là gian hàng khuyến mại. Khi đến gian hàng khuyến mại, khách hàng không những không phải trả một khoản phí nào mà còn có thể thực hiện tiếp đúng  $k$  bước di chuyển không mất tiền ngay sau đó. Chú ý rằng số bước di chuyển miễn phí là không tích lũy, nghĩa là khi tới một gian hàng khuyến mại, bất kể số bước di chuyển miễn phí còn hay hết, khách hàng cũng chỉ được đúng  $k$  bước di chuyển miễn phí ngay tiếp theo mà thôi.

Những cửa vào hội chợ được đặt ở những gian hàng nằm trên biên trái; còn những lối ra của hội chợ được đặt ở những gian hàng nằm trên biên phải. Từ một gian hàng bất kỳ có thể đi sang một trong những gian hàng chung cạnh với gian hàng đó bằng một bước di chuyển.

**Yêu cầu:** Hãy tìm một đường đi thăm hội chợ (đường đi từ một cửa vào tới một lối ra) sao cho tổng số tiền phải trả là ít nhất.

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

- Dòng 1: Chứa ba số nguyên  $m, n, k$  ( $1 \leq m \leq 1000; 2 \leq n \leq 1000; 1 \leq k \leq 5$ )
- $m$  dòng tiếp theo, dòng thứ  $i$  chứa  $n$  số tự nhiên, số thứ  $j$  là  $a_{ij}$  ( $0 \leq a_{ij} \leq 10000$ )

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

- Dòng 1: Ghi tổng số tiền phải trả.
- Các dòng tiếp theo mỗi dòng ghi chỉ số hàng và chỉ số cột của một ô trên đường đi. Thứ tự các ô được liệt kê trên những dòng này phải theo đúng thứ tự trên hành trình: Bắt đầu từ một cửa vào, kết thúc là một lối ra.

*Các số trên một dòng của Input / Output files được/phải ghi cách nhau ít nhất một dấu cách*

**Ví dụ:**

FAIR. INP	FAIR. OUT
6 7 2	14
1 5 1 1 1 17	2 1
4 0 7 7 7 12	2 2
9 9 2 2 1 10	2 3
9 10 10 10 1 10 10	2 4
9 10 10 10 1 2 3	3 4
9 10 10 10 10 10 10	3 5
	4 5
	5 5
	5 6
	5 7

## 8. DANH SÁCH VÒNG

Để làm việc với một danh sách gồm  $n$  số nguyên cần phải có hai thao tác:

- Thao tác Top chuyển phần tử đầu tiên của danh sách xuống vị trí cuối cùng của danh sách.
- Thao tác Bottom chuyển phần tử cuối cùng của danh sách lên vị trí đầu tiên của danh sách.

Một phép biến đổi danh sách đã cho là việc thực hiện  $k$  lần thao tác Top, rồi sau đó đến  $l$  lần thao tác Bottom.

Do số lần thực hiện phép biến đổi trên là rất lớn nên đòi hỏi phải có những thủ tục thực hiện hiệu quả để thực hiện liên tiếp  $x$  phép biến đổi đưa danh sách về trạng thái cuối cùng.

**Yêu cầu:** Viết chương trình cho phép với một danh sách và ba số  $k, l, x$  cho trước, xác định trạng thái của danh sách sau  $x$  lần thực hiện phép biến đổi.

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

- Dòng đầu tiên chứa ba số nguyên dương  $n, k, l$  ( $n, k, l \leq 10^5$ )
- Dòng thứ hai chứa  $n$  số nguyên, mỗi số có giá trị tuyệt đối không quá  $10^9$ , được sắp xếp theo thứ tự tương ứng với trạng thái khởi đầu của danh sách.
- Dòng thứ ba chứa số nguyên  $x$  ( $0 \leq x < 2^{64}$ )

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

Ghi ra trên một dòng của file văn bản CLIST.OUT các phần tử của danh sách sau  $x$  phép biến đổi. Các phần tử phải được ghi đúng thứ tự từ phần tử đầu tiên đến phần tử cuối cùng.

*Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách.*

**Ví dụ:**

CLIST.INP	CLIST.OUT
5 2 1	7 3 5 2 4
3 5 2 4 7	
9	