

## Khu vườn

Khu vườn của Tom chứa  $N$  đồng đất, đồng đất thứ  $i$  có  $A_i$  đơn vị đất. Trước khi trồng cây trong khu vườn Tom muốn có một sự thay đổi trong khu vườn của mình: đồng đất thứ  $i$  có  $B_i$  đơn vị đất. Để chỉnh sửa lại khu vườn, Tom có nhiều sự lựa chọn: Anh ta có thể mua một đơn vị đất và bỏ nó vào đồng đất mà anh ta muốn với giá  $X$  đồng. Anh ta có thể bỏ một đơn vị đất từ một đồng đất mà anh ta muốn với giá  $Y$  đồng. Anh ta cũng có thể di chuyển một đơn vị đất từ đồng đất thứ  $i$  sang đồng đất thứ  $j$  với giá  $Z * |i-j|$  đồng. Hãy giúp Tom tính số tiền nhỏ nhất mà anh ta phải bỏ ra để chỉnh sửa lại khu vườn.

**Dữ liệu:** vào từ file văn bản LANDS.INP

- Dòng 1: chứa 4 số tự nhiên  $N, X, Y$ , và  $Z$  ( $1 \leq N \leq 100, 0 \leq X, Y, Z \leq 1000$ ).
- Dòng thứ  $i$  trong  $N$  dòng tiếp theo chứa hai số tự nhiên là  $A_i$  và  $B_i$ . ( $A_i$  và  $B_i$  đều là số tự nhiên từ  $0..10$ )

**Kết quả:** ghi ra file văn bản LANDS.OUT

- Gồm một dòng chứa duy nhất một số nguyên ghi giá trị nhỏ nhất mà Tom cần phải trả để chỉnh sửa lại khu vườn.

**Ví dụ:**

LANDS.INP	LANDS.OUT
4 100 200	210
1	
1 4	
2 3	
3 2	
4 0	

## Robot

Một bảng hình chữ nhật kích thước là  $M \times N$ , trong đó các dòng được đánh số thứ tự từ 1 tới  $M$ , các cột được đánh số thứ tự từ 1 tới  $N$ . Trên mỗi ô ghi một số nguyên dương  $C[i,j]$  gọi là độ cao của ô  $(i,j)$ . Một Robot cần đi từ vị trí có tọa độ  $(1,1)$  tới vị trí có tọa độ  $(M,N)$ . Từ một ô  $(i,j)$  Robot có thể di chuyển sang ô  $(i+1,j)$  hoặc ô  $(i,j+1)$ , sao cho chênh lệch độ cao giữa hai ô đó là một số nguyên tố. Một ô  $(u,v)$  ( $1 < u < M, 1 < v < N$ ) được gọi là một ô đặc biệt nếu như mọi đường đi từ ô  $(1,1)$  tới ô  $(M,N)$  luôn phải qua ô  $(u,v)$ .

**Yêu cầu:** Viết chương trình tìm tất cả các ô đặc biệt.

**Dữ liệu:** vào từ file văn bản ROBOT.INP

- Dòng 1 ghi 2 số nguyên  $M$  và  $N$  ( $1 \leq M, N \leq 100$ )
- $M$  dòng tiếp theo, mỗi dòng ghi  $N$  số nguyên dương  $C[i,j]$  ( $C[i,j] \leq 100$ )

**Kết quả:** ghi ra file văn bản ROBOT.OUT

- Dòng 1 ghi số  $K$  là số lượng ô đặc biệt.
- $K$  dòng tiếp theo, mỗi dòng ghi cặp số  $u, v$  là tọa độ của ô đặc biệt.
- Tọa độ các ô đặc biệt phải được in ra theo thứ tự tăng dần của tọa độ dòng.

**Ví dụ:**

ROBOT.INP	ROBOT.OUT
5 6	2
1 4 8 6 2 9	2 2
6 9 12 15 8 4	4 5
8 12 2 4 13 4	
7 5 7 9 11 9	
8 6 8 9 14 16	

## Dãy số

Xét dãy số nguyên dương  $a = (a_1, a_2, \dots, a_n)$ . Giá trị của dãy là sự chênh lệch giữa số lớn nhất và số nhỏ nhất của dãy.

Ví dụ dãy  $(3, 1, 7, 2)$  có giá trị là 6, còn dãy  $(42, 42)$  có giá trị 0.

**Yêu cầu:** Cho  $N$  và dãy số  $a$ . Hãy tính tổng giá trị của tất cả các dãy con có không ít hơn 2 phần tử liên tiếp của  $a$ .

**Dữ liệu:** vào từ file văn bản SUM.INP:

- Dòng đầu tiên chứa số nguyên  $N$  ( $2 \leq N \leq 300000$ )
- Dòng thứ  $i$  trong  $N$  dòng sau chứa số nguyên  $a_i$ . ( $1 \leq a_i \leq 10^8$  với  $i = 1..N$ )

**Kết quả:** ghi ra file văn bản SUM.OUT một số nguyên là tổng tìm được.

**Ví dụ:**

SUM.INP	SUM.OUT
4	12
7	
5	
7	
5	

## Ếch săn mồi

Có  $m$  bậc thang đánh số từ 1 đến  $m$  từ trên xuống dưới. Mỗi bậc thang được chia đều thành  $n$  ô. Ô thứ  $j$  của bậc thang  $i$  được gọi là ô  $(i, j)$  và trên đó có lượng thức ăn  $a_{ij}$ .

Một con ếch muốn đi săn mồi trên những bậc thang. Ếch được xuất phát từ một ô tùy ý trên bậc thang 1 và nhảy dần xuống bậc thang  $m$ . Khi nhảy tới ô nào thì ếch sẽ ăn hết thức ăn trong ô đó. Tuy nhiên có một hạn chế là từ ô  $(x, y)$  chú ếch chỉ được phép nhảy sang ô  $(x', y')$  nếu:

$$\begin{cases} x' = x + 1 \\ |y' - y| \leq k \end{cases}$$

**Yêu cầu:** Tìm một cách đi kiếm ăn cho chú ếch sao cho tổng lượng thức ăn kiếm được là lớn nhất.

**Dữ liệu:** vào từ file văn bản FROG.INP:

- Dòng 1 chứa ba số nguyên dương  $m, n, k \leq 1000$
- $m$  dòng tiếp theo, dòng thứ  $i$  chứa  $n$  số nguyên dương, số thứ  $j$  là  $a_{ij} \leq 10^9$

**Kết quả:** ghi ra file văn bản FROG.OUT

- Dòng 1 ghi tổng lượng thức ăn kiếm được
- $m$  dòng tiếp theo, dòng thứ  $i$  ghi một số nguyên là số hiệu ô đi qua trên bậc thang  $i$ .

**Ví dụ:**

FROG.INP	FROG.OUT
3 5 2	18
4 3 2 1 1	3
4 3 5 4 9	5
1 2 3 7 5	4