

# BÀI TẬP SỐ 1

(Lưu ý tên file, file inp, file out là tên của đề bài)

## Bài 1. CETVRTA

Mirko cần chọn 4 điểm trong mặt phẳng mà nó có hình dạng một hình chữ nhật sao cho các cạnh của nó song song với các trục. Anh ấy đã chọn được cho mình và tự tin rằng anh ấy sẽ không sai, thế nhưng anh ta lại gặp vấn đề trong việc chọn điểm cuối, hãy giúp anh ấy.

### Input:

Mỗi điểm chọn trên một dòng. Mọi tọa độ là số nguyên dương thuộc khoảng từ 1 đến 100.

### Output:

Tọa độ điểm thứ 4 cần tìm thỏa mãn nó sẽ kết hợp với 3 điểm trên thành 1 hình chữ nhật.

Sample test:

Input:

5 5

5 7

7 5

Output:

7 7

## Bài 2. CETIRI

Mirko chọn 4 số nguyên sao cho sau khi sắp xếp chúng lại thì ta được 1 dãy tăng dần. Không may Mirko làm mất đi 1 số trong 4 số đó. Bạn hãy giúp Mirko tìm lại số đó nhé ( ͡° ͜ʖ ͡°)

Input

3 số nguyên từ -100 đến +100.

Lưu ý: Test luôn đảm bảo điều kiện bài toán.

Output

1 số tự nhiên thỏa mãn Mirko.

Example:

Input

4 6 8

Output

10

Input

10 1 4

Output

7

## Bài 3. TRI

Little Mirko viết vào trong vở một phương trình gồm 3 số nguyên , một dấu “=” và một trong 4 phép toán ( cộng + , trừ - , nhân \* và chia / ) . Nhưng bạn Slavko đã xóa mất phép toán và dấu bằng từ vở của Mirko .

Hãy giúp Mirko bằng cách khôi phục lại phương trình bằng 3 số nguyên .

### Input

Dòng thứ nhất gồm 3 số nguyên ( < 100) , cách nhau bởi dấu cách .

Chú ý : Dữ liệu nhập vào bảo đảm luôn có đáp số , mặc dù không nhất thiết là chỉ có một .

## **Output**

Trên một dòng duy nhất ,xuất ra một phương trình đúng gồm 3 số nguyên (đã cho ở trên) , dấu “=” và một trong 4 phép toán .

Nếu có nhiều cách giải , xuất ra một trong số đó .

Ví dụ

Input		Input
5 3 8		5 15 3
Output		Output
5+3=8		5=15/3

## **Bài 4. SKOCIMIS**

Ba con chuột túi đang chơi trên sa mạc. Chúng đang chơi trên một hàng gồm các số ,mỗi con chiếm giữ một số nguyên khác nhau. Trong mỗi lần di chuyển, một trong những con chuột túi bên ngoài nhảy vào khoảng trống giữa 2 con kia. Không có lúc nào 2 con chiếm cùng một vị trí.

**INPUT:** 3 số nguyên A,B,C (  $0 < A < B < C < 100$ ) là vị trí ban đầu của 3 con.

**OUTPUT:** Xuất ra số bước di chuyển lớn nhất mà chuột túi có thể thực hiện.

**EXAMPLES:**

<b>Input:</b> 2 3 5 <b>Output:</b> 1	<b>Input:</b> 3 5 9 <b>Output:</b> 3
---	---

## **Bài 5. KORNISLAV**

Chú rùa Kornislav không bao giờ có bất cứ điều gì thú vị để làm. Từ khi nó phải sống 300 năm, nó luôn cố tìm cách giết thời gian. Cuối tuần này, nó bắt đầu chơi “bao vây hình chữ nhật lớn nhất.”

Để bắt đầu, Kornislav cần bốn số nguyên dương. Nó cố gắng để rào một hình chữ nhật bằng cách di chuyển theo một hướng, sau đó quay 90 độ và đi theo hướng mới, vv. Kornislav thực hiện tổng cộng ba lượt 90 độ và đi bốn đoạn.

Khi đi theo một hướng nào đó, số bước nó thực hiện phải bằng một trong bốn số nguyên dương đã chọn và mỗi số nguyên chỉ được sử dụng duy nhất một lần. Tùy thuộc vào thứ tự mà Kornislav sử dụng các số nguyên, bước đi của nó sẽ tạo thành các hình dạng khác nhau, một số trong đó không chứa hình chữ nhật kèm theo.

Viết chương trình tính toán hình chữ nhật lớn nhất mà rùa có thể đi kèm với bước đi của nó.

**INPUT:** Dòng đầu tiên chứa bốn số nguyên dương A, B, C và D ( $0 < A, B, C, D < 100$ ), bốn số nguyên được chọn.

**OUTPUT:** Cho ra khu vực lớn nhất.

Ví Du:

Input	Output
1 2 3 4	3
4 4 3 4	12

Trong ví dụ đầu tiên, cách để Kornislav rào một hình chữ nhật của khu vực 3:

- Tiến lên 4 bước
- Rẽ phải
- Tiến lên 1 bước
- Rẽ phải
- Tiến lên 3 bước
- Rẽ phải
- Tiến lên 2 bước

## Bài 6. SPAVANAC

Đề thi khá dài nhưng nội dung chỉ gồm như sau:

Input: 1 dòng duy nhất gồm 2 số nguyên H và M ( $0 \leq H \leq 23$ ,  $0 \leq M \leq 59$ ) cách nhau bởi dấu cách, là thời gian hiện tại. H là giờ, M là phút.

Output:

1 dòng duy nhất gồm 2 số nguyên cách nhau bởi dấu cách là thời gian trước thời gian ở input 45 phút.

Vd:

10 10	0 30	23 40
9 25	23 45	22 55

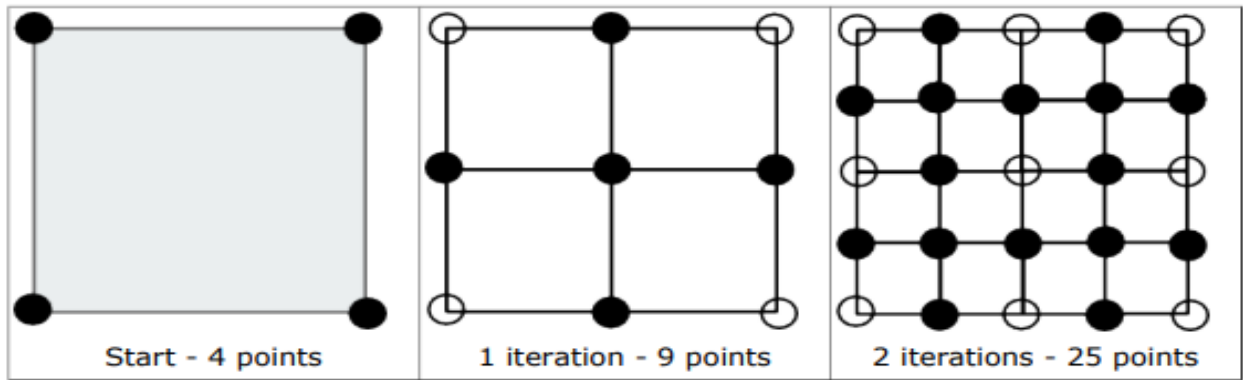
## Bài 7. PLANINA

Mirko và Slavko đang quay một bộ phim chuyển thể từ tiểu thuyết SF nổi tiếng "Chicks in Space 13". Kịch bản yêu cầu họ trình diễn ở nhiều nơi khác nhau trên thế giới nên họ quyết định quay toàn bộ phim trước phong xanh và thêm hình nền CGI sau. Mirko nghe nói rằng cách tốt nhất để tạo địa hình nhân tạo là sử dụng thuật toán thay điểm giữa (midpoint displacement algorithm).

Để bắt đầu thuật toán, Mirko chọn 4 điểm tạo thành một hình vuông hoàn hảo. Rồi anh thực hiện các bước sau:

1. Ở mỗi cạnh của hình vuông, anh ta thêm một điểm mới ở trung điểm của cạnh đó. Chiều cao của điểm mới này là chiều cao trung bình của hai điểm trên cạnh đó.
2. Ở trung tâm của hình vuông, anh ta thêm một điểm mới có chiều cao là chiều cao trung bình của cả 4 đỉnh vuông, cộng với một giá trị nhỏ ngẫu nhiên.

Sau khi hai bước được thực hiện, bây giờ anh ta có 4 hình vuông mới. Anh ấy thực hiện các bước như vậy trên các ô vuông mới được tạo ra nhiều lần cho đến khi anh ta hài lòng với kết quả. Sơ đồ sau minh họa 2 lần lặp của thuật toán.



Mirko nhận thấy rằng một số điểm thuộc nhiều hơn một hình vuông. Để giảm sự tiêu thụ bộ nhớ, anh ta lưu trữ tính toán và lưu trữ các điểm như vậy chỉ một lần. Bây giờ anh ta tự hỏi mình sẽ cần bao nhiêu điểm trong bộ nhớ sau N lần lặp đi lặp lại.

#### INPUT

Dòng đầu tiên và duy nhất chứa một số nguyên N ( $1 \leq N \leq 15$ ), số lần lặp đi lặp lại.

#### OUTPUT

Dòng duy nhất chứa một số, là số điểm được lưu trữ sau N lần lặp.

### Bài 8. SOK

Mirko và Slavko mua một vài lít nước cam, táo và dưa. Các bạn ấy đang định làm cocktail không cồn theo công thức tìm được trên mạng. Nhưng thật không may, các bạn đã tìm ra quá muộn. Họ không nên chỉ dùng công thức để làm cocktail mà nên dùng để lên kế hoạch mua nước quả nữa.

Viết chương trình để xác định từng loại nước quả mà họ sẽ phải bỏ thừa, sau khi họ làm nhiều cocktail nhất có thể theo công thức.

#### Input

Dòng thứ nhất bao gồm 3 số nguyên A, B, C ( $1 \leq A, B, C \leq 500$ ), tương ứng là số lít nước cam, táo và dưa mà họ đã mua.

Dòng thứ 2 bao gồm I, J, K ( $1 \leq I, J, K \leq 50$ ), tương ứng là số lít nước cam, táo và dưa cần để làm một ly cocktail.

#### Output

Trên một dòng xuất ra duy nhất gồm 3 số là phần thừa của mỗi loại nước quả (lít).

Ví dụ :

Input :	Input :	Input :
10 10 10	9 9 9	10 15 18
3 3 3	3 2 1	3 4 1
Output:	Output:	Output:
0 0 0	0 3 6	0 1.666667 14.666667

--	--	--

## Bài 9. CUDOVISTE

Mirko được bố mẹ thưởng một chiếc xe bán tải khổng, kích thước của nó bằng 4 chiếc xe cỡ nhỏ. Vì kích thước xe khổng nên Mirko khó khăn trong việc tìm vị trí đỗ xe. Bạn của Mirko đã cho Mirko một tấm bản đồ thành phố, bản đồ có R dòng và C cột, mỗi ô trong bảng chứa một trong các ký tự '#', '.' hoặc 'X' với ký tự '#' cho biết vị trí đó có toà nhà, ký tự 'X' cho biết vị trí đó đã có xe nhỏ đăng ký và ký tự '.' cho biết vị trí đó còn trống. Xe của Mirko sẽ cần vị trí 2x2 ô để đậu, anh ấy có thể đậu xe trên vị trí trống hoặc trên vị trí mà người khác đã đăng ký nhưng không thể đậu xe đè lên toà nhà.

Yêu cầu: Hãy giúp Mirko đếm số vị trí đậu xe.

Dữ liệu vào:

- Dòng đầu chứa 2 số R và C ( $2 \leq R, C \leq 50$ )
- R dòng sau mỗi dòng chứa C ký tự thể hiện bản đồ thành phố.

Kết quả: Đưa ra số vị trí đậu xe ứng với số vị trí bị xe nhỏ đăng ký chỗ:

- Dòng thứ nhất 0 xe.
- Dòng thứ hai là 1 xe
- Dòng thứ ba 2 xe.
- Dòng thứ tư 3 xe
- Dòng cuối cùng 4 xe

Ví dụ:

Input	Output	Input	Output	Input	Output
4 4	1	4 4	90000	4 5	2
#..#	1	....		..XX.	1
..X.	2	....		..XX.	1
..X.	1	....		..#..	0
#XX#	0	....		.....	1

## Bài 10. NOIXICH

Người ta có N đoạn dây xích ( $N \leq 20000$ ), mỗi đoạn dây xích là một chuỗi các mắt xích được nối với nhau. Các đoạn dây xích này tách rời nhau. Mỗi đoạn dây xích có không quá 20000 mắt xích.

Bằng cách cắt ra một mắt xích, sau đó hàn lại, ta có thể nối hai dây xích thành một đoạn. Thời gian cắt và hàn mỗi mắt xích là 1 đơn vị thời gian và được xem là bằng nhau với mọi mắt xích

Nhiệm vụ của bạn là phải nối chúng lại thành một đoạn dây xích duy nhất với thời gian ít nhất (hay số mắt xích bị cắt ra và hàn lại là ít nhất)

**Dữ liệu:** Dữ liệu được cho trong file NOIXICH.INP có cấu trúc như sau:

- Dòng đầu tiên ghi số N là số đoạn mắt xích
- Những dòng tiếp theo ghi N số nguyên dương, số thứ i là số mắt xích có trong đoạn thứ i ( $1 \leq i \leq N$ ). Hai số cạnh nhau trên một dòng cách nhau ít nhất một dấu cách.

**Kết quả:** Chương trình của bạn cần đưa ra file NOIXICH.OUT một số duy nhất là số đơn vị thời gian để bạn cần để nối N đoạn xích đã cho.

*Ví dụ:*

NOIXICH.INP	NOIXICH.OUT
3 4 7 6	2
4 1 2 3 4	2