

# [Set Map]. Bài 1. Phần tử phân biệt

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho dãy số  $A[]$  gồm có  $N$  phần tử, bạn hãy đếm xem trong mảng có bao nhiêu phần tử phân biệt? Chú ý giải bài này với 3 cách : Sử dụng set, map, sắp xếp.

Nếu dùng set hoặc map bạn chỉ cần đưa các phần tử vào set hoặc đưa phần tử vào làm key trong map thì nó sẽ tự loại bỏ trùng, khi đó size của set và map sẽ là số lượng phần tử khác nhau trong mảng  $A[]$  ban đầu. 3 cách dùng set, map, sort đều có độ phức tạp là  $O(N\log N)$  nên bạn dùng cách nào cũng được

## Input Format

Dòng đầu tiên là số nguyên  $N$ . Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$

## Constraints

$$1 \leq N \leq 10^5; -10^9 \leq A[i] \leq 10^9$$

## Output Format

In ra số lượng phần tử khác nhau trong mảng.

## Sample Input 0

```
5
2 1 2 1 3
```

## Sample Output 0

```
3
```

# [Set Map]. Bài 2. Truy vấn phần tử trong mảng.

- **Problem**
- [Submissions](#)
- [Discussions](#)

Cho dãy số  $A[]$  gồm có  $N$  phần tử, bạn hãy trả lời các truy vấn để xác định xem phần tử  $X$  nào đó có nằm trong mảng hay không? Bạn hãy thử giải bài này bằng 3 cách : Set, Map, Binary Search.

Nếu dùng set, map để lưu mảng  $A[]$  mỗi truy vấn tìm kiếm (hàm find) bạn chỉ mất độ phức tạp  $O(\log N)$  thay vì  $O(N)$  như duyệt tuần tự, tương tự bạn có thể sort mảng  $A[]$  để áp dụng tìm kiếm nhị phân trên mỗi truy vấn.

### Input Format

- Dòng đầu tiên là số nguyên  $N$ .
- Dòng thứ 2 gồm  $N$  số nguyên  $A[i]$
- Dòng thứ 3 là số lượng truy vấn  $Q$ .
- $Q$  dòng tiếp theo mỗi dòng là một số nguyên  $X$ .

### Constraints

$1 \leq N \leq 10^5$ ;  $0 \leq A[i], X \leq 10^9$ ;  $1 \leq Q \leq 1000$ ;

### Output Format

Đối với mỗi truy vấn in ra YES nếu phần tử  $X$  xuất hiện trong mảng, ngược lại in ra NO.

### Sample Input 0

```
5
1 2 1 3 4
2
1
6
```

### Sample Output 0

```
YES
NO
```

# [Set Map]. Bài 3. Phần tử riêng biệt.

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho dãy số  $A[]$  gồm có  $N$  phần tử, bạn hãy liệt kê các giá trị xuất hiện trong dãy theo thứ tự xuất hiện, mỗi giá trị chỉ liệt kê một lần.

Gợi ý : Dùng set để lưu các phần tử đã được in ra trước đó, gặp phần tử  $a[i]$  thì kiểm tra nhanh trong set có giá trị  $a[i]$  chưa để in ra

```
set<int> se;
for(int i = 0; i < n; i++){
    //Nếu a[i] chưa nằm trong set thì in ra
    //Check trong set sẽ nhanh hơn vì chỉ mất  $O(\log N)$ 
    //Thêm a[i] vào set
}
```

## Input Format

Dòng đầu tiên là số nguyên  $N$ . Dòng thứ 2 gồm  $N$  số nguyên  $A[i]$

## Constraints

$1 \leq N \leq 10^5$ ;  $0 \leq A[i], X \leq 10^9$ ;

## Output Format

In ra các giá trị xuất hiện trong mảng theo thứ tự xuất hiện.

## Sample Input 0

```
8
1 2 3 2 1 3 4 8
```

## Sample Output 0

```
1 2 3 4 8
```

# [Set Map]. Bài 4. Truy vấn trên mảng

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho dãy số  $A[]$  gồm có  $N$  phần tử, có 3 thao tác như sau :

- Thao tác 1 : Thêm 1 phần tử  $X$  vào mảng.
- Thao tác 2 : Xóa 1 phần tử  $X$  khỏi mảng. Trong trường hợp phần tử  $X$  không xuất hiện trong mảng, sẽ không thực hiện xóa, nếu trong mảng có nhiều phần tử  $X$  thì chỉ xóa đi 1 phần tử  $X$  trong mảng.
- Thao tác 3 : Truy vấn xem phần tử  $X$  có xuất hiện trong mảng hay không?

Ban đầu mảng  $A$  có  $N$  phần tử, với các truy vấn phần tử  $X$  trong mảng, in ra YES nếu  $X$  xuất hiện trong mảng, ngược lại in ra NO.

Gợi ý : Bài này vừa muốn xóa nhanh vừa muốn tìm kiếm nhanh lại phải lưu được giá trị trùng nhau => Sử dụng multiset

```
multiset<int> se;
for(int i = 0; i < q; i++){
    //Nhập thao tác
    //Thao tác 1 => Thêm vào se
    //Thao tác 2 => Vì xóa 1 phần tử nên cần tìm kiếm rồi
    //xóa qua iterator
    //Thao tác 3 dùng hàm find
}
```

### Input Format

- Dòng đầu tiên là số nguyên  $N$ .
- Dòng thứ 2 gồm  $N$  số nguyên  $A[i]$ .
- Dòng thứ 3 là số lượng thao tác  $Q$ .
- $Q$  dòng tiếp theo mỗi dòng là thông tin của một thao tác, gồm 2 số, số đầu tiên là loại thao tác, số tiếp theo là phần tử  $X$  trong thao tác.

### Constraints

$1 \leq N \leq 10^4$ ;  $0 \leq A[i], X \leq 10^9$ ;  $1 \leq Q \leq 1000$ ;  $0 \leq X \leq 10^9$ ;

## Output Format

Đối với mỗi thao tác loại 3, in ra YES nếu X xuất hiện trong mảng, ngược lại in ra NO.

## Sample Input 0

```
6
1 2 3 4 5 6
4
1 1
2 6
3 1
3 6
```

## Sample Output 0

```
YES
NO
```

## Explanation 0

Ban đầu mảng có 6 phần tử 1, 2, 3, 4, 5, 6. Sau thao tác đầu tiên mảng có thêm 1 phần tử 1 và trở thành mảng (1, 1, 2, 3, 4, 5, 6). Sau thao tác thứ 2, mảng xóa đi 1 phần tử 6 và trở thành mảng (1, 1, 2, 3, 4, 5). Thao tác thứ 3 truy vấn phần tử 1, câu trả lời là YES vì 1 xuất hiện trong mảng. Thao tác thứ 4 truy vấn phần tử 6, câu trả lời là NO vì 6 không xuất hiện trong mảng.

# [Set Map]. Bài 5. Tần suất ký tự

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho N kí tự được nhập từ bàn phím, bạn hãy thực hiện các yêu cầu sau

- 1. In ra kí tự có từ điển nhỏ nhất và tần suất xuất hiện của nó
- 2. In ra kí tự có từ điển lớn nhất và tần suất xuất hiện của nó
-

3. In ra các kí tự xuất hiện trong N kí tự đã cho theo thứ tự từ điển tăng dần kèm tần suất của nó

•

4. In ra các kí tự xuất hiện trong N kí tự đã cho theo thứ tự từ điển giảm dần kèm tần suất của nó

### Input Format

- Dòng 1 là N : số lượng kí tự
- Dòng 2 gồm N kí tự viết cách nhau 1 dấu cách

### Constraints

- $1 \leq N \leq 10^5$

### Output Format

- In ra kết quả của 4 yêu cầu, mỗi yêu cầu viết cách nhau 1 dòng

### Sample Input 0

```
20
E a q A z A F F A f G r r l p g D p n M
```

### Sample Output 0

```
A 3
z 1

A 3
D 1
E 1
F 2
G 1
M 1
a 1
f 1
g 1
l 1
n 1
p 2
q 1
r 2
z 1

z 1
r 2
q 1
p 2
n 1
```

```
l 1
g 1
f 1
a 1
M 1
G 1
F 2
E 1
D 1
A 3
```

## [Set Map]. Bài 6. Distinct pair

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho N điểm trong hệ tọa độ Oxy, bạn hãy xác định có bao nhiêu điểm riêng biệt trong các điểm đã cho. Gợi ý : Dùng set hoặc map lưu pair

### Input Format

- Dòng 1 là N : số lượng điểm đã cho
- N dòng tiếp theo mỗi dòng gồm hoành độ và tung độ của 1 điểm

### Constraints

- $1 \leq N \leq 10^6$
- $-10^6 \leq x, y \leq 10^6$

### Output Format

In ra số lượng điểm khác nhau

### Sample Input 0

```
20
2 2
0 1
1 2
2 1
2 0
2 1
2 2
```

```
0 0
2 0
0 2
0 1
1 1
0 2
0 2
1 1
0 2
1 2
0 1
1 1
1 1
```

### Sample Output 0

```
8
```

## [Set Map]. Bài 8. Lớn nhất, nhỏ nhất

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng  $A[]$  gồm  $n$  phần tử, có 4 loại thao tác

- Thao tác 1 : Thêm phần tử  $X$  vào mảng
- Thao tác 2 : Xóa mọi giá trị  $X$  khỏi mảng nếu  $X$  tồn tại trong mảng
- Thao tác 3 : Tìm phần tử nhỏ nhất trong mảng
- Thao tác 4 : Tìm phần tử lớn nhất trong mảng

Đối với thao tác 3 và 4 bạn hãy in ra phần tử lớn nhất, nhỏ nhất trong mảng  $A[]$

### Input Format

- Dòng 1 gồm  $N$  : Số lượng phần tử ban đầu trong mảng  $A[]$
- Dòng thứ 2 gồm  $N$  phần tử ban đầu của mảng  $A[]$
- Dòng 3 gồm số lượng thao tác  $Q$



- Q dòng tiếp theo mỗi dòng mô tả 1 thao tác, trong đó nếu là thao tác 1 và 2 thì có thêm phần tử X.

### Constraints

- $1 \leq N \leq 10^5$
- $1 \leq Q \leq 10000$
- $0 \leq A[i], X \leq 10^6$

### Output Format

In ra kết quả với thao tác 3 và 4

### Sample Input 0

```
10
24 31 49 65 57 77 7 81 77 79
10
2 60
2 7
3
3
2 45
1 16
4
3
1 6
1 8
```

### Sample Output 0

```
24
24
81
16
```

## [Set Map]. Bài 9. Union

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng A[] và B[] lần lượt gồm N và M số nguyên, nhiệm vụ của bạn là tìm hợp của 2 mảng này và in theo thứ tự từ lớn về nhỏ.

Gợi ý : Để tìm hợp của 2 mảng thì chỉ cần dùng 1 set và lưu mọi phần tử của 2 mảng, set sẽ tự loại bỏ các phần tử trùng nhau trong cả 2 mảng.

Cú pháp duyệt ngược của set như sau :

```
for(auto it = se.rbegin(); it != se.rend(); ++it){  
    cout << *it << ' ';  
}
```

### Input Format

- Dòng 1 gồm N và M
- Dòng 2 gồm N số trong mảng A[]
- Dòng 3 gồm M số trong mảng B[]

### Constraints

- $1 \leq N, M \leq 10^5$
- $-10^9 \leq A[i], B[i] \leq 10^9$

### Output Format

In ra tập hợp của 2 mảng theo thứ tự giảm dần

### Sample Input 0

```
9 6  
7 9 2 0 0 5 6 4 7  
9 7 3 4 5 7
```

### Sample Output 0

```
9 7 6 5 4 3 2 0
```

## [Set Map]. Bài 21. Set và kiểu char

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng A[] gồm N phần tử là ký tự, bạn hãy đếm xem có bao nhiêu ký tự khác nhau trong mảng. Gợi ý : Để in theo thứ tự từ điển bạn chỉ cần

duyệt set, còn in theo thứ tự ngược các bạn có thể dùng iterator ngược hoặc đưa các phần tử trong set ra vector rồi in từ cuối vector về đầu.

### Input Format

- Dòng 1 là N : số lượng phần tử trong mảng
- Dòng thứ 2 gồm N ký tự trong mảng

### Constraints

- $1 \leq N \leq 10^6$
- Các ký tự là chữ cái hoặc chữ số

### Output Format

- Dòng 1 in ra số lượng ký tự khác nhau trong mảng
- Dòng 2 in ra các ký tự khác nhau theo thứ tự từ điển tăng dần
- Dòng 3 in ra các ký tự khác nhau theo thứ tự từ điển giảm dần

### Sample Input 0

```
12
2 8 h t t 2 h e h e h t
```

### Sample Output 0

```
5
2 8 e h t
t h e 8 2
```

## [Set Map]. Bài 22. Số lần xuất hiện

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng A[] gồm N phần tử, nhiệm vụ của bạn đó là với mỗi phần tử trong mảng bạn hãy chỉ ra đó là lần thứ mấy nó xuất hiện ? Ví dụ mảng A[] = {1, 2, 3, 1, 1, 4, 2, 3} sẽ có kết quả là : 1, 1, 1, 2, 3, 1, 2, 2

Gợi ý : Cách 1 các bạn dùng multiset và hàm count. Cách 2 là dùng map.

### Input Format

- Dòng 1 là N : số lượng phần tử trong mảng
- Dòng 2 là N số cách nhau 1 dấu cách

### Constraints

- $1 \leq N \leq 10^6$
- $-10^9 \leq A[i] \leq 10^9$

### Output Format

In ra kết quả của bài toán

### Sample Input 0

```
10
1 0 1 0 4 0 1 3 3 0
```

### Sample Output 0

```
1 1 2 2 1 3 3 1 2 4
```

## [Set Map]. Bài 10. Intersection

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng A[] và B[] lần lượt gồm N và M số nguyên, nhiệm vụ của bạn là tìm giao của 2 mảng này và in theo thứ tự xuất hiện trong mảng A[].

Gợi ý :

Cách 1 : Dùng 2 set để lưu 2 mảng A, B sau đó duyệt 1 trong 2 mảng và tìm kiếm sự xuất hiện của phần tử đang duyệt trong mảng còn lại.

Cách 2 : Dùng 1 map để đánh dấu, duyệt qua các phần tử trong mảng A[] và cho map của từng phần tử trong mảng A[] = 1, duyệt qua mảng B[] thì

kiểm tra nếu map của phần tử trong mảng B đã bằng 1 thì chuyển thành 2. Cuối cùng duyệt map và in ra những phần tử được đánh dấu là 2.

### Input Format

- Dòng 1 gồm N và M
- Dòng 2 gồm N số trong mảng A[]
- Dòng 3 gồm M số trong mảng B[]

### Constraints

- $1 \leq N, M \leq 10^5$
- $-10^9 \leq A[i], B[i] \leq 10^9$

### Output Format

In ra tập hợp của 2 mảng theo thứ tự xuất hiện trong mảng A[]

### Sample Input 0

```
9 8
7 5 0 0 1 6 2 7 3
6 4 9 4 2 8 3 5
```

### Sample Output 0

```
5 6 2 3
```

## [Set Map]. Bài 11. Difference

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng A[] và B[] lần lượt gồm N và M số nguyên, nhiệm vụ của bạn là tìm những phần tử xuất hiện trong mảng thứ nhất nhưng không xuất hiện trong mảng thứ 2, liệt kê theo thứ tự từ bé đến lớn.

Gợi ý : Dùng 2 set để lưu mảng A, B. Duyệt các phần tử trong set 1 và tìm kiếm nếu phần tử ko nằm trong set 2 thì in ra.

### Input Format

- Dòng 1 gồm N và M
- Dòng 2 gồm N số trong mảng A[]
- Dòng 3 gồm M số trong mảng B[]

### Constraints

- $1 \leq N, M \leq 10^5$
- $-10^9 \leq A[i], B[i] \leq 10^9$

### Output Format

In ra đáp án của bài toán

### Sample Input 0

```
6 6
1 9 3 8 0 7
3 1 7 3 7 5
```

### Sample Output 0

```
0 8 9
```

## [Set Map]. Bài 25. Set và phần tử thứ 2

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng A[] gồm N phần tử, gọi B[] là mảng chứa các phần tử khác nhau trong mảng A[], bạn hãy in ra : - Số lượng phần tử của mảng B

- Giá trị lớn nhất, nhỏ nhất của mảng B[]
- Giá trị lớn thứ nhì, nhỏ thứ nhì của mảng B[]

Dữ liệu đảm bảo B[] có ít nhất 2 phần tử.

### Input Format

- Dòng 1 là N : số lượng phần tử trong mảng

- Dòng 2 là N số cách nhau 1 dấu cách

### Constraints

- $1 \leq N \leq 10^6$
- $-10^9 \leq A[i] \leq 10^9$

### Output Format

In ra 3 dòng theo yêu cầu

### Sample Input 0

```
17
1 0 9 1 5 1 4 7 9 7 5 2 6 3 3 4 8
```

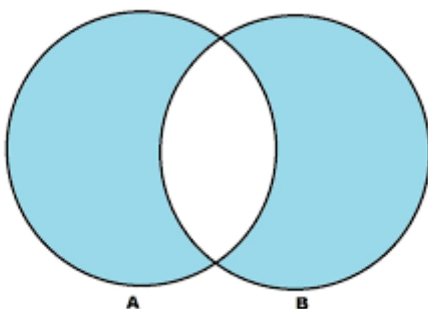
### Sample Output 0

```
10
9 0
8 1
```

## [Set Map]. Bài 12. Symmetric difference

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng A[] và B[] lần lượt gồm N và M số nguyên, nhiệm vụ của bạn là tìm những phần tử xuất hiện ở 1 trong 2 mảng nhưng không được xuất hiện ở cả 2 mảng.



### Input Format

- Dòng 1 gồm N và M
- Dòng 2 gồm N số trong mảng A[]
- Dòng 3 gồm M số trong mảng B[]

### Constraints

- $1 \leq N, M \leq 10^5$
- $-10^9 \leq A[i], B[i] \leq 10^9$

### Output Format

In ra đáp án của bài toán theo thứ tự từ bé đến lớn

### Sample Input 0

```
9 8
9 0 2 3 9 7 8 1 4
1 0 2 1 1 2 0 5
```

### Sample Output 0

```
3 4 5 7 8 9
```

## [Set Map]. Bài 13. Loại trừ

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng A[] và B[] có N và M phần tử là các số nguyên, hãy tìm các phần tử xuất hiện trong mảng 1 mà không xuất hiện trong mảng 2.

### Input Format

Dòng đầu tiên là N và M; Dòng thứ 2 là N số của mảng A[]; Dòng thứ 3 là M số của mảng B[];

### Constraints

$1 \leq N, M \leq 10^6$ ;  $-10^9 \leq A[i], B[i] \leq 10^9$ ;

### Output Format



In ra các số xuất hiện trong mảng 1 mà không xuất hiện trong mảng 2 theo thứ tự từ bé đến lớn, mỗi giá trị thỏa mãn chỉ liệt kê 1 lần.

### Sample Input 0

```
3 4
1 2 3
1 2 4 5
```

### Sample Output 0

```
3
```

## [Set Map]. Bài 14. Suffix And Query

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng  $A[]$  gồm  $N$  phần tử. Bạn được yêu cầu trả lời cho các truy vấn, mỗi truy vấn yêu cầu bạn đếm số các giá trị khác nhau từ chỉ số  $L$  tới chỉ số  $N - 1$ .

Gợi ý : Gọi  $F[i]$  là số lượng các giá trị khác nhau tính từ chỉ số  $i$  tới chỉ số  $n - 1$  trong mảng, xây dựng trước mảng  $F[i]$ , sau đó mỗi truy vấn chỉ cần truy cập vào  $F[l]$  là có thể tìm được số lượng các giá trị khác nhau từ  $l$  tới  $n - 1$ . Cách xây dựng mảng  $F[]$  là duyệt từ cuối về và dùng 1 set để lưu các giá trị khác nhau tính từ cuối, gán  $F[i] = \text{set.size}()$

### Input Format

- Dòng đầu tiên là  $N$  - số lượng phần tử của mảng
- Dòng thứ 2 gồm  $N$  phần tử  $A[i]$  của mảng
- Dòng thứ 3 là số truy vấn  $Q$
- $Q$  dòng tiếp theo mỗi dòng là một số  $L$ .

### Constraints

- $1 \leq N \leq 1000000$

- $1 \leq A[i] \leq 10^9$
- $1 \leq Q \leq 10000$
- $0 \leq L \leq N-1$

### Output Format

Với mỗi truy vấn in kết quả trên 1 dòng.

### Sample Input 0

```
9
3 3 3 4 2 0 3 1 2
4
8
6
1
7
```

### Sample Output 0

```
1
3
5
2
```

## [Set Map]. Bài 15. Set vs Lower\_bound, Upper\_bound

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho 1 mảng  $A[]$  gồm  $N$  số nguyên, 28tech cung cấp cho bạn các thao tác sau :

- 1 x : Thêm phần tử  $x$  vào mảng  $A[]$
- 2 x : Xóa phần tử  $x$  khỏi mảng  $A[]$  nếu  $x$  tồn tại trong mảng, và nếu  $x$  xuất hiện nhiều lần bạn chỉ xóa đi 1 phần tử duy nhất.
- 3 x : Chỉ ra giá trị đầu tiên của phần tử  $\geq x$  trong  $A[]$  khi mảng  $A[]$  được sắp xếp tăng dần, nếu không tồn tại in ra -1

- 4 x : In ra giá trị của phần tử lớn nhất  $\leq x$  trong mảng A[], nếu không tồn tại phần tử này thì in ra -1.

### Input Format

- Dòng 1 là N
- Dòng 2 là N phần tử A[i]
- Dòng 3 là số lượng thao tác T
- T dòng tiếp theo mô tả thao tác.

### Constraints

- $1 \leq N \leq 1000$
- $1 \leq A[i], X \leq 1000$ ;
- $1 \leq T \leq 1000$

### Output Format

Đối với thao tác 3 và 4, in ra kết quả tương ứng

### Sample Input 0

```
9
6 4 9 3 8 6 8 1 6
9
4 3
4 6
3 6
3 4
1 6
2 6
2 7
4 4
1 4
```

### Sample Output 0

```
3
6
6
4
4
```

# [Set Map]. Bài 17. Truy vấn kí tự

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho một xâu kí tự  $S$  chỉ bao gồm các kí tự từ  $a$  tới  $z$ . Bạn được yêu cầu thực hiện các nhiệm vụ sau bằng CTDL phù hợp. - 1. Tìm kí tự có tần suất xuất hiện nhiều nhất và có thứ tự từ điển nhỏ nhất.

- 2. Tìm kí tự có tần suất xuất hiện nhỏ nhất và có thứ tự từ điển lớn nhất.
- 3. Tìm số lượng kí tự khác nhau trong xâu.

## Input Format

Xâu kí tự  $S$ .

## Constraints

$1 \leq \text{len}(S) \leq 10^6$ ;

## Output Format

- Dòng đầu tiên in ra kí tự có tần suất lớn nhất, nếu có nhiều kí tự có cùng tần suất in ra kí tự có thứ tự từ điển nhỏ nhất.
- Dòng thứ 2 in ra kí tự có tần suất nhỏ nhất, nếu có nhiều kí tự có cùng tần suất thì in ra kí tự có thứ tự từ điển lớn nhất.
- Dòng 3 in ra số lượng kí tự khác nhau trong xâu.

## Sample Input 0

```
mghzbzhrucvwjssatuhlwcradwtbmyqdjcksyskvhrgcubvtoimrjwnanfqgenshrjtf
```

## Sample Output 0

```
h 5
o 1
```

# [Set Map]. Bài 7. Phần tử xuất hiện ở mọi dòng

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho 1 bảng số gồm N dòng và mỗi dòng gồm N số, nhiệm vụ của bạn là in ra những giá trị xuất hiện ở mọi dòng. Gợi ý : Dùng map đánh dấu, dòng 1 : đánh dấu các giá trị ở dòng 1 có value = 1, dòng 2 sẽ kiểm tra giá trị nào được đánh dấu ở dòng 1 mới đánh dấu tại dòng 2, tương tự tới dòng n, những số nào được đánh dấu ở cả n dòng sẽ thỏa mãn yêu cầu.

## Input Format

- Dòng 1 là N : số dòng của bảng số
- N dòng tiếp theo mỗi dòng gồm N số nguyên viết cách nhau 1 dấu cách

## Constraints

- $1 \leq N \leq 100$
- Các số trong bảng số là số nguyên int

## Output Format

- In ra các số xuất hiện ở mọi dòng theo thứ tự từ nhỏ đến lớn, nếu không tồn tại số nào in ra NOT FOUND

## Sample Input 0

```
5
0 1 4 0 4
3 2 2 3 0
0 3 4 0 1
0 1 4 4 1
0 0 0 0 2
```

## Sample Output 0

```
0
```

# [Set Map]. Bài 23. Liệt kê tần suất chẵn

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng  $A[]$  gồm  $N$  phần tử, bạn hãy in ra các giá trị có số lần xuất hiện chẵn trong mảng.

## Input Format

- Dòng 1 là  $N$  : số lượng phần tử trong mảng
- Dòng 2 là  $N$  số cách nhau 1 dấu cách

## Constraints

- $1 \leq N \leq 10^6$
- $-10^9 \leq A[i] \leq 10^9$

## Output Format

- Đầu tiên bạn in ra theo thứ tự tăng dần các giá trị sau đó cách ra 1 dòng và in ra theo thứ tự giảm dần về các giá trị

## Sample Input 0

```
11
2 2 1 2 4 4 3 2 4 0 0
```

## Sample Output 0

```
0 2
2 4

2 4
0 2
```