

A - ASCII code

Los valores ASCII de las letras minúsculas en inglés 'a', 'b', ..., 'z' son 97, 98, ..., 122 en ese orden. Dado un entero N entre 97 y 122, muestre por pantalla la letra correspondiente cuyo valor ASCII es N .

Restricciones

- N es un entero entre 97 y 122.

Input

Una línea que contiene un único entero N .

Output

Muestre por pantalla la letra correspondiente cuyo valor ASCII.

Ejemplo

Input

```
1 97
```

Output

```
1 a
```

```
1 122
```

Output

```
1 z
```

B - Jugando con estructuras

Un amigo y tú están asistiendo a las clases de OCILabs, luego de la clase de pilas y colas tu amigo se siente un poco confundido. Este te dice que no entiende la diferencia entre una pila y cola.

Tú, como buen amigo, quieres programar un código que reciba una secuencia de números y muestre el orden en que salgan tanto de una pila como de una cola.

Input

La primera línea contiene un entero n ($1 \leq n \leq 105$) - corresponda la cantidad de números dentro de la secuencia.

Luego le sigue una línea que contiene n enteros, los que corresponden a los números de la secuencia, cada entero tendrá un valor entre 1 y 104.

Output

Debe imprimir una línea que diga "PILA: " y luego los números en el orden que son sacados de la pila, separados por un espacio.

Luego debe seguir otra línea que diga "COLA: " y luego los números en el orden que son sacados de la cola, separados por un espacio.

Ejemplo

Input

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

Output

```
1 PILA: 4 3 2 1
2 COLA: 1 2 3 4
```

Input

```
1 1
2 10
```

Output

```
1 PILA: 10
2 COLA: 10
```

C - Bibliotecas compartidas

Un grupo de amigos quiere abrir una biblioteca, llamada OCIBiblioteca, para ello deciden cada uno agregar libros de su propia colección. Luego, ellos quieren hacer un listado de todos los diferentes libros que poseen en común.

Input

La primera línea contiene un entero n ($1 \leq n \leq 1000$) - correspondiente a la cantidad de personas que van a compartir sus libros.

Luego le siguen n líneas, donde cada una posee un entero l_i ($1 \leq l_i \leq 1000$) - correspondiente a la cantidad de libros de la persona i . Luego del entero le siguen l_i nombres de libros, estos nombres contienen solo letras minúsculas del abecedario inglés.

Output

Debe mostrar por pantalla los nombres de todos los diferentes libros que poseen en común, cada uno en una línea diferente.

Ejemplo

Input

```
1 3
2 4 pilas colas vectores conjuntos
3 2 conjuntos maps
4 1 conjuntos
```

Output

```
1 colas
2 conjuntos
3 maps
4 pilas
5 vectores
```

NOTA:

En el ejemplo, existen tres personas. La primera persona posee los libros **pilas**, **colas**, **vectores** y **conjuntos**. La segunda persona posee los libros **conjuntos** y **maps**. La tercera persona posee los libros **conjuntos**. Por lo tanto, los nombres de todos los diferentes libros que poseen en común son: **colas**, **conjuntos**, **maps**, **pilas** y **vectores**.

D - A Reverse

Le dan los números enteros L , R y un string S que consta de letras minúsculas en inglés.

Muestre por pantalla después de invertir (el orden de) los caracteres L -ésimo a R -ésimo.

Restricciones

- S consiste de letras en minúsculas del abecedario ingles.
- $1 \leq |S| \leq 10^5$ ($|S|$ corresponde al largo del string S)
- L y R son enteros $1 \leq L \leq R \leq |S|$

Input

La primera línea contiene los eenteros L y R . Luego le sigue otra línea que contiene el string S .

Output

Mostrar por pantalla el string según lo explicado en el enunciado.

Ejemplo

Input

```
1 3 7
2 abcdefgh
```

Output

```
1 abgfedch
```

```
1 1 7
2 reviver
```

Output

```
1 reviver
```

```
1 4 13
2 merrychristmas
```

Output

```
1 meramtsirhcyr
```

E - Good Sequence

Te dan una secuencia de números enteros positivos de longitud N , $a=(a_1, a_2, \dots, a_n)$. Su objetivo es eliminar algunos de los elementos en a para que sea una buena secuencia.

Aquí, una secuencia b es una buena secuencia cuando se cumple la siguiente condición:

Para cada elemento x en b , el valor x ocurre exactamente x veces en b .

Por ejemplo, $(3, 3, 3)$ (se encuentre tres veces el número 3), $(4, 2, 4, 1, 4, 2, 4)$ (se encuentra una vez el número 1, se encuentra dos veces el número 2, se encuentra cuatro veces el número 4) y $()$ (una secuencia vacía) son buenas secuencias, mientras que $(3, 3, 3, 3)$ (se encuentra cuatro veces el número 3) y $(2, 4, 1, 4, 2)$ no lo son.

Encuentre la cantidad mínima de elementos que deben eliminarse para que a sea una buena secuencia.

Restricciones

- $1 \leq N \leq 10^5$
- a_i es un entero
- $1 \leq a_i \leq 10^9$

Input

La primera línea contiene el entero N . Luego, la siguiente línea contiene N números separados por un espacio.

Output

Debe mostrar por pantalla el número mínimo de elementos que deben eliminarse para que sea una buena secuencia.

Ejemplo

Input

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

Output

```
1 1
```

Explicación: para que sea una buena secuencia se debe eliminar un 3.

Input

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

Output

1 2

Explicación: para que sea una buena secuencia se debe eliminar los dos 4.

Input

1 6
2 1 2 2 3 3 3

Output

1 0

Explicación: no se debe eliminar nada es una buena secuencia.

Input

1 1
2 1000000000

Output

1 1

Explicación: para que sea una buena secuencia se debe eliminar el 1000000000, así queda una secuencia vacía y por lo tanto una buena secuencia.

Input

1 8
2 2 7 1 8 2 8 1 8

Output

1 5