



> A - RegistrOCCI

El problema consiste en implementar un sistema de registro de usuarios para un nuevo servicio de correo electrónico llamado "RegistrOCCI".

Si un usuario desea registrarse con un nombre, el sistema verifica si ese nombre ya está en la base de datos.

Las instrucciones son las siguientes:

- Si no existe, el nombre se inserta en la base de datos y el usuario recibe una respuesta "OK".
- Si el nombre ya existe, el sistema genera un nuevo nombre agregando números al final del nombre original (nombre1, nombre2, ...) y lo inserta en la base de datos, y luego envía el nuevo nombre como una sugerencia al usuario.

El número que se agrega comienza en 1 y se incrementa hasta encontrar el primer número disponible que no está en uso en la base de datos.

* Input:

La primera línea contiene un número n ($1 \leq n \leq 10^5$). Le siguen n líneas que contiene consultas al sistema. Cada consulta es una línea no vacía y consiste de no más de 32 caracteres, de los cuales todos son letras minúsculas del alfabeto inglés.

* Output:

Imprime el elemento mínimo distinto de cero antes de cada operación en una nueva línea.

* Ejemplo

Input:

```
4
abacaba
acaba
abacaba
acab
```

Output:

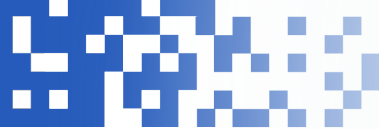
```
OK
OK
abacaba1
OK
```

Input:

```
6
first
first
second
second
third
third
```

Output:

```
OK
first1
OK
second1
OK
third1
```



> B - Apilar palitos

Martina ha recibido un kit de construcción. El kit consta de varios palitos de madera, cuyas longitudes son conocidas. Los palitos pueden ser apilados uno encima del otro si sus longitudes son iguales.

Martina quiere construir el número mínimo de torres con los palitos. Ayuda a Martina a utilizar los palos de la mejor manera posible.

* Input:

La primera línea contiene un entero N ($1 \leq N \leq 1000$) — el número de palitos a disposición de Martina.

La segunda línea contiene N enteros separados por espacios, l_i que son las longitudes de los palitos. Todas las longitudes son números naturales que no exceden los 1000.

* Output:

En una línea, imprime dos números — la altura de la torre más alta y la cantidad de pilas que logra crear Martina. Recuerda que Martina debe usar todos los palos.

* Ejemplo

Input:

```
3
1 2 3
```

Output:

```
1 3
```

Input:

```
4
6 5 6 7
```

Output:

```
2 3
```



> C - Árboles y sus porcentajes

Imagínate que tienes un mapa con información sobre cada árbol en una zona determinada. Queremos saber cuántos de esos árboles son de cada tipo (por ejemplo, de hojas anchas o de agujas) y qué porcentaje representan de todos los árboles en el área. Así podemos aprender más sobre los tipos de árboles que hay allí.

Te piden que respondas en orden alfabético, cual es el árbol y su porcentaje con respecto a la cantidad total de arboles.

Para calcular el porcentaje de un árbol, divida la cantidad de veces que aparece ese tipo de árbol por la cantidad total de arboles que hay, luego multiplique ese resultado por 100.

* Input:

La primera línea indica el número de casos de prueba, seguida de una línea en blanco. Cada caso de prueba consiste en una lista de especies de árboles observados por satélite, con un árbol por línea. Ningún nombre de especie supera los 30 caracteres. Hay un máximo de 10,000 especies y 1,000,000 árboles en cada caso. Hay una línea en blanco entre cada par de casos de prueba.

Nota: Recuerda que puedes usar la función `getline` de C++ para obtener toda la línea

```
Consola:
#include<iostream>
#include<string>
using namespace std;

int main() {
    string s;
    // esto lee la linea completa
    getline(cin, s);
    cout << s << endl;
}
```

* Output:

Para cada caso de prueba, imprimir el nombre de cada especie representada en la población, en orden alfabético, seguido del porcentaje de la población que representa, con 4 decimales.

También debes imprimir una línea en blanco entre dos casos de prueba consecutivos.

* Ejemplo

Input:

```
3

Red Alder
Ash

Ash
Beech
Yellow Birch
Ash

Cherry
```



Output:

```
Ash 50.0000
Red Alder 50.0000

Ash 50.0000
Beech 25.0000
Yellow Birch 25.0000

Cherry 100.0000
```

Explicación:

En el primer ejemplo:

1. El árbol tipo Ash aparece 1 vez, por lo tanto el porcentaje es $1 \cdot 100/2 = 50.0000$
2. El árbol tipo Red Alder aparece 1 vez, por lo tanto el porcentaje es $1 \cdot 100/2 = 50.0000$

En el segundo ejemplo:

1. El árbol tipo Ash aparece 2 vez, por lo tanto el porcentaje es $2 \cdot 100/4 = 50.0000$
2. El árbol tipo Beech aparece 1 vez, por lo tanto el porcentaje es $1 \cdot 100/4 = 25.0000$
3. El árbol tipo Yellow Birch aparece 1 vez, por lo tanto el porcentaje es $1 \cdot 100/4 = 25.0000$

En el tercer ejemplo, hay solo 1 tipo de árbol por lo que ese árbol tiene porcentaje 100.0000

Tip

Para imprimir con precisión a 4 decimales, realice lo siguiente:

```
Consola:

#include<iostream>
#include<string>
#include <iomanip>
using namespace std;

int main() {
    float f = 123.12312323;
    // esto imprimirá f con 4 decimales
    cout << fixed << setprecision(4) << endl;
}
```