

Galletas para los gemelos

Nombre del problema	Galletas para los gemelos
Archivo de entrada	entrada estándar
Archivo de salida	salida estándar
Tiempo límite	1 segundo
Memoria límite	256 megabytes

Este es un problema interactivo. Tu programa se comunicará con nuestro calificador escribiendo mensajes en la salida estándar y leyendo mensajes de la entrada estándar de manera alterna.

Sophie está preparando la fiesta de cumpleaños de sus gemelos. Los gemelos aman las galletas. Por su cumpleaños, ellos quieren probar algo nuevo: galletas de la Compañía de Galletas Dulzor Único (CGDU).

Cada galleta producida por la CGDU tiene un valor de dulzor entero entre 1 y 10^{16} (incluyendo los extremos). Dado que los gemelos de Sophie siempre se ponen celosos uno del otro, cada uno de ellos debe recibir galletas con la misma suma de valores de dulzor.

La CGDU solo acepta órdenes de **exactamente** n galletas. En cada orden, el cliente especifica el valor de dulzor de cada una de las n galletas que va a pedir.

Manteniéndose fiel a su nombre, la Compañía de Galletas Dulzor Único se rehusa a producir dos galletas con el mismo valor de dulzor para el mismo cliente. Sophie debe asegurarse de nunca ordenar el mismo valor de dulzor dos veces -- ni en la misma orden, ni entre todas ellas. Sophie nunca ha comprado en la CGDU anteriormente, así que ella puede ordenar cada posible valor de dulzor a lo mucho una vez.

Hay un obstáculo más en el camino de Sophie: Es sabido que el servicio de entrega a domicilio de la CGDU es terrible. Cada vez que un cliente ordena n galletas, solo una de esas n galletas termina llegando al cliente. El resto de las galletas es comido por los empleados del servicio de entrega durante el trayecto. El cliente no puede influenciar en cuál de las n galletas ordenadas le llegará al final.

Ya que el cumpleaños se acerca rápidamente, Sophie tiene tiempo para hacer a lo mucho 101 órdenes. Tu tarea es ayudarla.

Más específicamente, debes hacer lo siguiente:

1. Primero, ordenar las galletas. Puedes hacer a lo mucho 101 órdenes, cada una de ellas con exactamente n valores de dulzor deseados. Debes hacer una orden a la vez. **Inmediatamente después de realizar cada orden, recibirás el valor de dulzor de la galleta que recibiste.**

Recuerda que no puedes usar el mismo valor de dulzor múltiples veces, incluso entre todas las órdenes. (En particular, si ordenaste una galleta con un valor de dulzor t pero esta no te llegó, **no puedes** ordenar una galleta con el mismo valor de dulzor de nuevo.)

2. Luego, dividir las galletas. Una vez que has recibido suficientes galletas, debes distribuir **algunas** de las galletas recibidas entre los gemelos. Ambos gemelos deben recibir al menos una galleta y la suma de valores de dulzor de las galletas que reciba el primer gemelo debe ser igual a la del segundo. **¡No es necesario que uses todas las galletas que recibiste!**

Salida

Cada vez que tu programa imprima una o más líneas en la entrada estándar, esta acción debe ser seguida por un **flush en el stream de salida**. Esto es necesario para asegurarte de que la información que imprimas le llegue al calificador inmediatamente.

Ejemplos de cómo se puede hacer esto:

- En C++, hay múltiples opciones.
 - `fflush(stdout);`
 - `std::cout << std::flush;`
 - `std::cout << std::endl;` (note que esto también imprime una nueva línea)
 - leer usando `std::cin` también hace flush en el stream de salida.
- En Java, puedes usar `System.out.flush()`
- En Python, puedes usar `sys.stdout.flush()`

Protocolo de Interacción

Tu programa debe ejecutar la siguiente secuencia de acciones:

1. Leer el valor n de la entrada estándar.
2. A lo mucho 101 veces:
 1. Primero, escribe una línea describiendo una orden de n galletas en la salida estándar.
 2. Luego, lee de la entrada estándar el valor de dulzor de la galleta que

recibiste. Está garantizado que el valor que leerás está entre los n valores de la orden que hiciste.

3. Imprime tres líneas describiendo una forma válida de dar a los gemelos algunas galletas de las que recibiste.

El calificador escribirá cada entero en una línea nueva.

Para ordenar galletas, imprime una sola línea con n seguido por n enteros: los valores de dulzor de las galletas que deseas ordenar. Imprime un solo espacio antes de cada uno de los n enteros.

Recuerda que puedes hacer a lo mucho 101 órdenes y que no tienes permitido ordenar el mismo valor de dulzor dos veces.

Una vez que hayas ordenado suficientes galletas, imprime tres líneas finales que describan qué galletas debe darle Sophie a cada uno de los gemelos.

La primera de estas líneas debe tener la forma " $m k$ " con $m, k > 0$: el número de galletas que debe recibir el primer y el segundo gemelo, respectivamente.

La segunda de estas líneas debe contener m enteros separados por un solo espacio: los valores de dulzor de las galletas que debe recibir el primer gemelo.

De manera similar, la tercera de estas líneas debe contener k enteros separados por un solo espacio: los valores de dulzor de las galletas que debe recibir el segundo gemelo.

La salida debe satisfacer las siguientes condiciones:

1. Cada gemelo debe recibir al menos una galleta.
2. La suma de valores de dulzor de las galletas del primer gemelo debe ser igual a la del segundo.
3. Solo puedes usar las galletas que recibiste de cada orden.
4. Cada una de las galletas que recibiste solo pueden ser dadas a lo mucho a uno de los gemelos.

Cualquier salida que satisfaga estas condiciones será aceptada. En particular, puedes imprimir las galletas seleccionadas en cualquier orden.

Luego de imprimir las tres líneas finales, haz un flush en el stream de salida una última vez y luego **termina tu programa normalmente**.

Puntajes

Subgrupo 1 (8 puntos): $n = 1$

Subgrupo 2 (9 puntos): $1 \leq n \leq 2$

Subgrupo 3 (18 puntos): $1 \leq n \leq 25$

Subgrupo 4 (16 puntos): $1 \leq n \leq 200$

Subgrupo 5 (13 puntos): $1 \leq n \leq 1000$

Subgrupo 6 (36 puntos): $1 \leq n \leq 5000$

Ejemplos

entrada estándar	salida estándar
1	? 13
13	? 7
7	? 31
31	? 12
12	? 5
5	? 3
3	! 2 3
	7 13
	12 5 3
2	? 3 7
7	? 2 8
2	? 1 5
5	! 2 1
	2 5
	7

Nota

Los ejemplos de entrada y salida deben ser leídos línea por línea. Tu programa lee un valor de la entrada estándar e imprime una línea (o tres líneas al final) en la salida estándar alternadamente.

El calificador decide qué galleta devolver arbitrariamente. Esto significa que el calificador puede ser adaptativo a tus consultas en algunos casos, pero también puede elegir galletas aleatoriamente en otros. En particular, para $n = 2$, si haces la misma secuencia de órdenes que en el segundo ejemplo, podrías recibir un conjunto de galletas diferente.