



Notas

Para todas las tareas:

- Los límites están disponibles en la página "Overview" en el sistema de la competencia.
- Hay un paquete adjunto que usted puede descargar del sistema de la competencia que contiene evaluadores de ejemplo, ejemplos de implementaciones, casos de prueba de ejemplo, así como scripts para compilar y ejecutar las soluciones.
- Usted puede hacer hasta 50 envíos por cada tarea, y debe enviar exactamente un archivo en cada envío.
- Cuando pruebe sus programas con el evaluador de ejemplo, su entrada debe coincidir con el formato y las restricciones especificadas en el enunciado del problema, de otra manera, pueden ocurrir comportamientos inesperados.
- En las entradas del evaluador de ejemplo, entre cada dos tokens en una línea existe un simple espacio que los separa, a menos que otro formato sea explícitamente indicado.
- Cuando esté probando su código en la computadora, le recomendamos que utilice los scripts proporcionados en los paquetes adjuntos. Por favor note que utilizamos la opción de compilador `-std=gnu++17`.
- Si usted no puede enviar el código al CMS, usted puede utilizar la herramienta `ioisubmit` para guardar su código
 - Ejecute `ioisubmit <nombre_corto_problema> <archivo_fuente>` en el directorio del `<archivo_fuente>`.
 - Solicite a un miembro del comité organizador que tome una foto de la salida que genere `ioisubmit`. Su entrega no será considerada a menos que realice este paso.
 - Si usted compite online, solicite a su encargado que tome una foto de la salida de `ioisubmit` y la envíe a los organizadores.

Convención

Los enunciados de los problemas especifican firmas utilizando nombres de tipo genérico `void`, `int`, `int64`, `int[]` (arreglos), and `int[][]` (arreglo de arreglos). En C++, los evaluadores utilizan los tipos de dato apropiados o implementaciones como se listan a continuación:

<code>void</code>	<code>int</code>	<code>int64</code>	<code>int[]</code>
<code>void</code>	<code>int</code>	<code>long long</code>	<code>std::vector<int></code>

<code>int[][]</code>	longitud del arreglo a
<code>std::vector<std::vector<int>></code>	<code>a.size()</code>