



Nota de implementación

- Debes enviar exactamente un archivo (el nombre del archivo está dado en el enunciado del problema).
- Este archivo implementa el subprograma descrito en el enunciado del problema usando las firmas (declaraciones de funciones/métodos) dadas en las plantillas de ejemplo.
- Estos subprogramas deben comportarse como se describe en el enunciado del problema.
- Eres libre de implementar otros subprogramas (funciones, procedimientos, métodos).
- Tus envíos no deben interactuar de ninguna manera con la entrada/salida estándar, ni leer o escribir archivos. En particular, si tu programa imprime cualquier cosa a la salida estándar, obtendrá el resultado SV (Security Violation). Puedes escribir cualquier cosa a la salida de error estándar.

Convenciones

Los enunciados de los problemas y las secciones de detalles de implementación usan algunos nombres de tipo genéricos, en particular:

- el nombre *arreglo* y el tipo correspondiente `int[]`
- el tipo `int64`
- el tipo `string`
- el tipo `boolean`

En cada uno de los lenguajes de programación permitidos los graders usan el tipo de dato apropiado de ese lenguaje como se muestra a continuación:

Lenguaje	arreglo	int64	string	boolean
C++	<code>std::vector<int></code>	<code>long</code> <code>long</code>	<code>std::string</code>	<code>bool</code>
C	<code>int*</code>	<code>long</code> <code>long</code>	<code>char*</code>	<code>int</code>
Pascal	<code>array of longint</code>	<code>int64</code>	<code>string</code>	<code>boolean</code>
Java	<code>int[]</code>	<code>long</code>	<code>String</code>	<code>boolean</code>

.
. .
. .
. .

Límites

Problema	Límite de tiempo	Límite de memoria
Paint By Numbers	2 segundos	2 GB
Unscrambling a Messy Bug	2 segundos	2 GB
Aliens	2 segundos	2 GB