



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 usan la palabra *arreglo* la sección de detalles de implementación usa el tipo `int[]`. Dependiendo del lenguaje de programación, los graders usan los siguientes tipos de datos en vez de `int[]` (y por simplicidad, los llamamos a todos arreglos):

- `std::vector<int>` en C++,
- `int*` en C,
- `array of longint` en Pascal,
- `int[]` en Java.

La sección de detalles de implementación usa el tipo `int64`. Este corresponde a un entero con signo de 64-bits:

- `long long` en C++/C,
- `int64` en Pascal,
- `long` en Java.

Límites

| Tarea | Tiempo límite | Memoria límite |
|-------------------------|---------------|----------------|
| Detecting Molecules | 1 segundo | 2 GB |
| Roller Coaster Railroad | 2 segundos | 2 GB |
| Shortcut | 2 segundos | 2 GB |