



Notice

Si estas desabilitado para enviar una solución vía CMS, tienes que hacer lo siguiente:

- Coloca tu archivo de envío en el directorio llamado `submit_<task name>` en su escritorio (el directorio ya debería existir) **antes que finalizar el concurso**.
 - Para problemas de única-salidas, los archivos deben llamarse `01.out`, `02.out`, ..., `10.out`.
 - Para otros problemas, debe haber **exactamente un** archivo en ese directorio, que contenga su código fuente.
- Pregunta al líder de tu equipo para enviar apelaciones. Si cree que se debe dar mas tiempo extra, haga lo siguiente:
- Enviar la solicitud de aclaración (por medio del sistema del concurso o en papel) lo antes posible.
- No abandones tu escritorio o hables con otros concursantes después de que termine el concurso. después de que termine el concurso.

Cada tarea tiene un paquete adjunto que tu puedes descargar desde el sistema del concurso.. Hay algunas tareas única-salida, para los cuales:

- Los archivos adjuntos contienen casos de prueba de entrada y ejemplo de casos de prueba. Cada caso de prueba esta subdividido por subtareas.
- Puedes enviar multiples archivos de salida como un archivo zip. Para este propósito, tu archivo de salida se debe llamado `?? .out`, donde `??` es el número de caso de prueba (ejemplo, `03.out`). Puedes usar muchos archivos zip con el siguiente comando: `zip output.zip *.out`
- Puedes realizar 100 envíos para tareas de única-salida. En cada envío, puedes enviar archivos de salida para cualquier subconjunto de los casos de prueba.

Para otras tareas:

- Los paquetes adjuntos contienen ejemplos de graders, ejemplos de implementaciones, ejemplo de casos de prueba y script compilados.
- Tu puedes realizar hasta 50 envíos para cada tarea, y tiene que enviar exactamente un archivo en cada envío.
- El nombre del archivo que usted debe enviar se proporciona en el encabezado de la declaración de tarea. Debe implementar los procedimientos descritos en la declaración de la tarea usando las firmas provistas en las implementaciones de

muestra.

- Usted es libre de realizar otros procedimientos.
- Sus envíos no deben leer desde la entrada estandar, escritos en la salida estandar, o interactuar con algunos archivos. Sin embargo, ellos pueden sacar por el standard error stream.
- tus envíos **no deben llamar exit() or System.exit()**
- Cuando este probando tus programas con el ejemplo grader, tus entradas deben coincidir con el formato y las restricciones de la declaración de la tarea, de lo contrario, puede ocurrir comportamientos no especificados.
- En los ejemplos de entradas grader, cada dos tokens consecutivos en una línea estan separados por un simple espacio, a menos que se especifique explícitamente otro formato.
- Cuando pruebes tu código en su computadora local, le recomendamos que use el script en el paquete adjunto. De lo contrario, especialmente en C++, agregar la opción `-std = gnu ++ 14` para compilar.

Convenciones

Las declaraciones de tareas especifican firmas utilizando nombres de tipo genérico `void`, `int`, `int64`, and `int[]` (array).

En cada uno de los lenguajes soportados, el graders usa apropiados tipos de datos o implementaciones, listados a continuación:

Language	void	int	int64	int[]	length of array a
C++	void	int	long long	<code>std::vector<int></code>	<code>a.size()</code>
Java	void	int	long	<code>int[]</code>	<code>a.length</code>

Limites

Task	Time limit	Memory limit
line	output-only	output-only
vision	1 sec	1024 MB
walk	4 sec	1024 MB