

Desarrolla un programa en C que almacene **dos listas de números** en memoria: Una lista de números **negativos** y otra de números **positivos**.

Tendrás que hacer uso de datos de tipo estructura para crear las listas. Todos los hilos deberán añadir números a las listas, teniendo en cuenta que éstas son compartidas por todos ellos.

Al finalizar el programa, se deberán pintar ambas listas por pantalla, indicando el número almacenado en cada posición de la lista.

### Condiciones a cumplir:

1. Todos los hilos deben ejecutarse en paralelo y todos deben intentar rellenar la lista lo más rápido posible.
2. Los números deberán ser de tipo INT.
3. Todos los números tienen que ser generados aleatoriamente y deben estar entre INT\_MIN e INT\_MAX.
4. El programa no debe tener memory leaks ni dar segfault.
5. El código deberá estar dividido en varias funciones y ficheros .c y .h.
6. Los errores se han de pintar en la salida standard de error. El resto de los datos a pintar, se deberán pintar en la salida estándar.
7. Es obligatorio que los números en las listas estén almacenados de forma consecutiva.
8. No es obligatorio acompañar el código con tests unitarios que validen su funcionamiento, aunque se valorará positivamente.

### Parámetros del programa

El programa debe recibir dos parámetros de entrada. Si no se proporciona ningún parámetro o se introduce un parámetro incorrecto, el programa debe

cerrarse limpiamente mostrando un mensaje de error y explicando el uso correcto.

1. **Primer parámetro:** Número total de hilos.
2. **Segundo parámetro:** Cantidad de números que generara cada hilo.

### Validaciones del programa

El programa debe verificar que:

1. Los parámetros de entrada sean correctos.
2. Los números positivos y negativos se gestionan por separado.
3. Deberá imprimir ambas listas en orden ascendente.

### Instrucciones:

1. **Implementa** el código en C.
2. **Asegúrate** de manejar correctamente la memoria y los errores.
3. **Repositorio:** el programa debe estar creado en un repositorio de GIT con commits que expliquen cada cambio realizado.

**Tiempo estimado:** 2 horas

**Nota:** El tiempo está muy ajustado a propósito dado que se busca valorar los criterios de decisión bajo estrés y/o escasez de tiempo. Por favor, intenta acotar la prueba al tiempo marcado.

