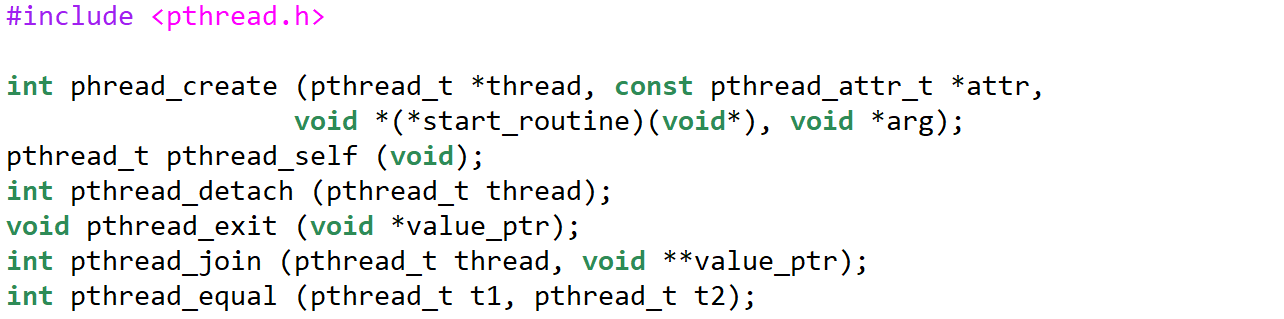
Hilos POSIX

# Creación de hilos



**pthread\_create** crea un nuevo hilo con los atributos attr dentro del proceso que la invoca y devuelve a través de **thread** su identificador. Si **attr** vale NULL se pasan los atributos por defecto. En caso de fallo la función devuelve un númeo distinto de 0 que indica el error producido. El hilo ejecuta el código de la rutina referenciada por **start\_routine** a la que se le pasan los argumentos referenciados por **arg**.

**pthread\_self** devuelve el identificador del hilo que la invoca.

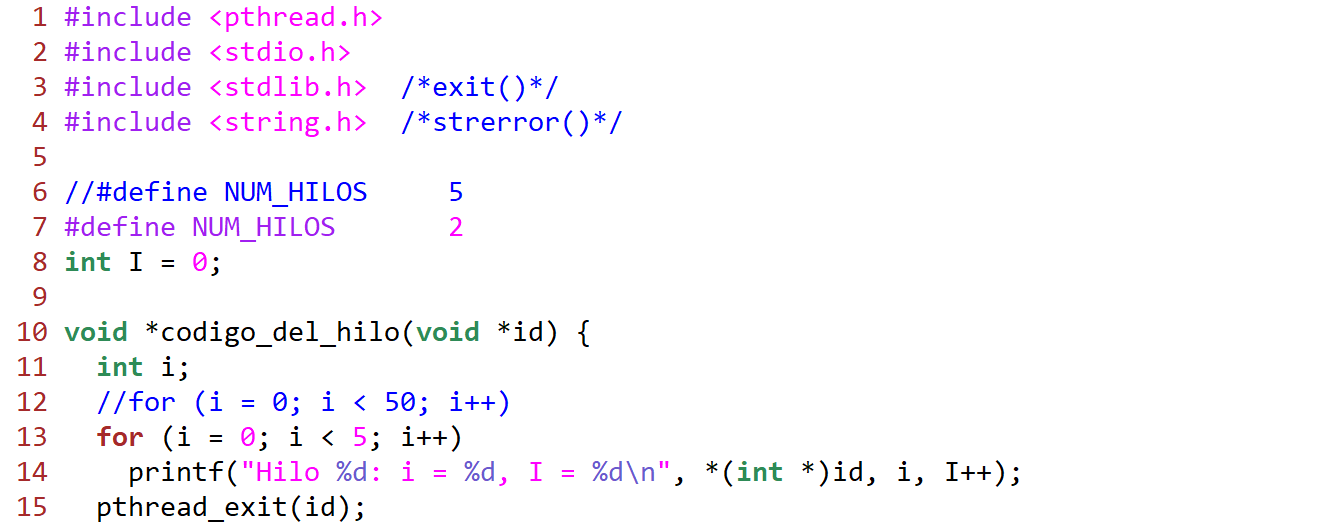
**pthread\_detach** le indica al sistema que los recursos de almacenamiento ocupados por el hilo con identificador **thread** tienen que ser liberados cuando el hilo termine.

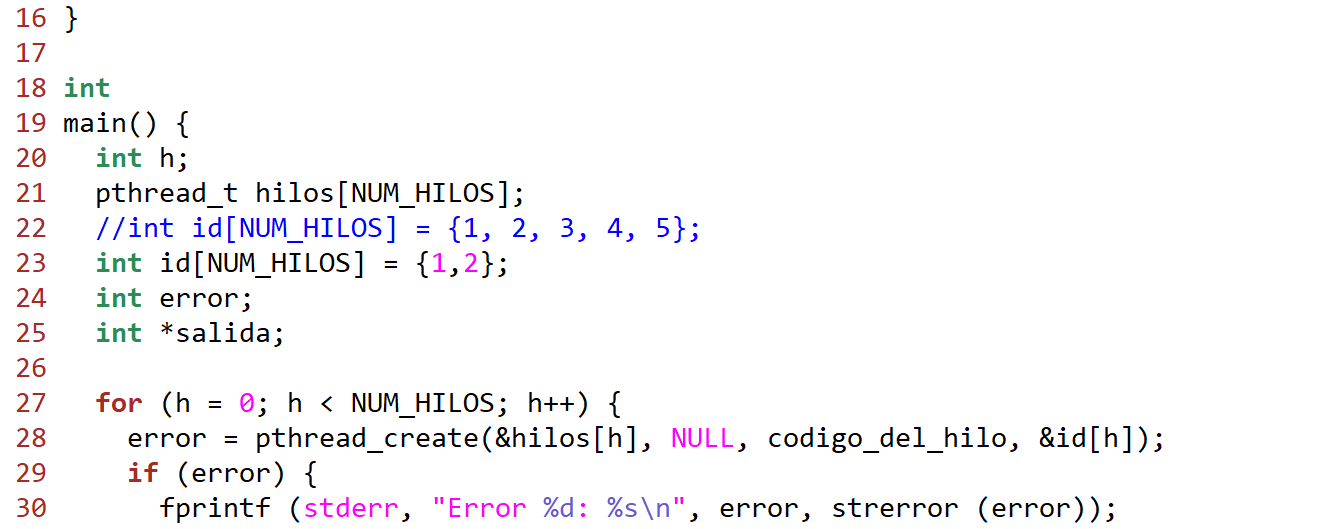
**pthread\_exit** termina la ejecución del hilo que la invoca y le devuelve los datos referenciados por **value\_ptr** al hilo que está esperando su terminación. Esta función estmabién llamada de forma implícita cuando termina la rutina **start\_routine** indicada al crear el hilo.

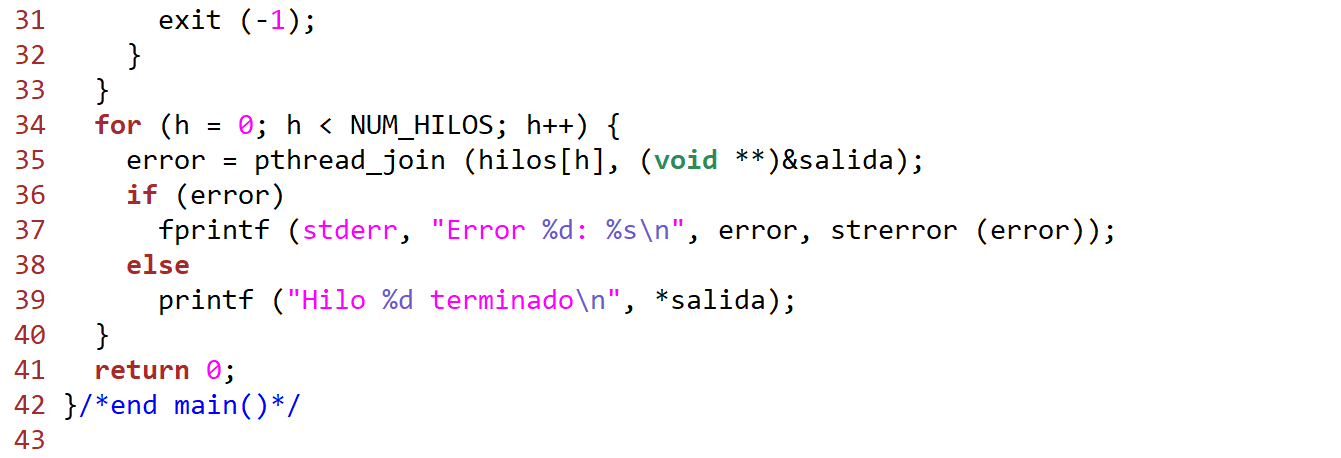
**pthread\_join** suspende la ejecución del hilo que la invoca y espera hasta la terminación del hilo con identificador **thread**. El apuntador devuelto por este hilo con una llamada a **pthread\_exit** es almacenado en la dirección referenciada por **value\_ptr**. Si un hilo intenta ejecutar pthread\_join sobre sí mismo se detecta un abrazo mortal y la llamada a la función falla devolviendo el valor **EDEADLK**.

**pthread\_equal** devuelve 0 si los identificadores **t1** y **t2** no son iguales y cualquier valor distinto de 0 si coinciden.

Considere el siguiente ejemplo, en el cual se crean varios hilos dentro de un proceso. El código de la función main también es otro hilo --llamado el hilo principal--. En el ejemplo, el hilo principal se encarga de esperar la terminación de los otros hilos.





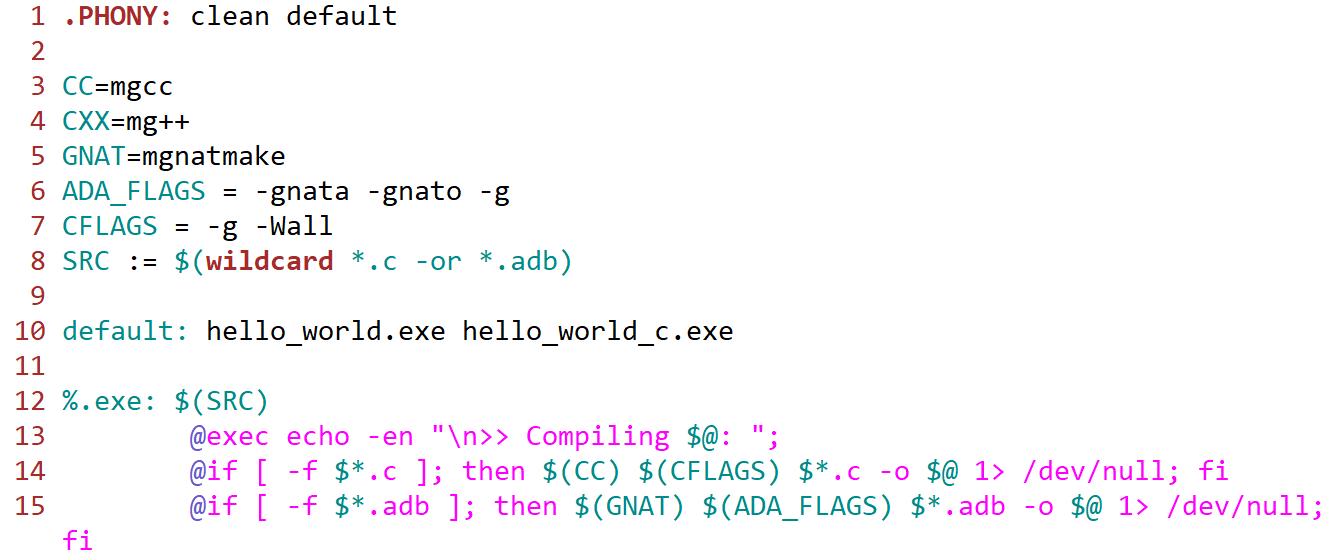


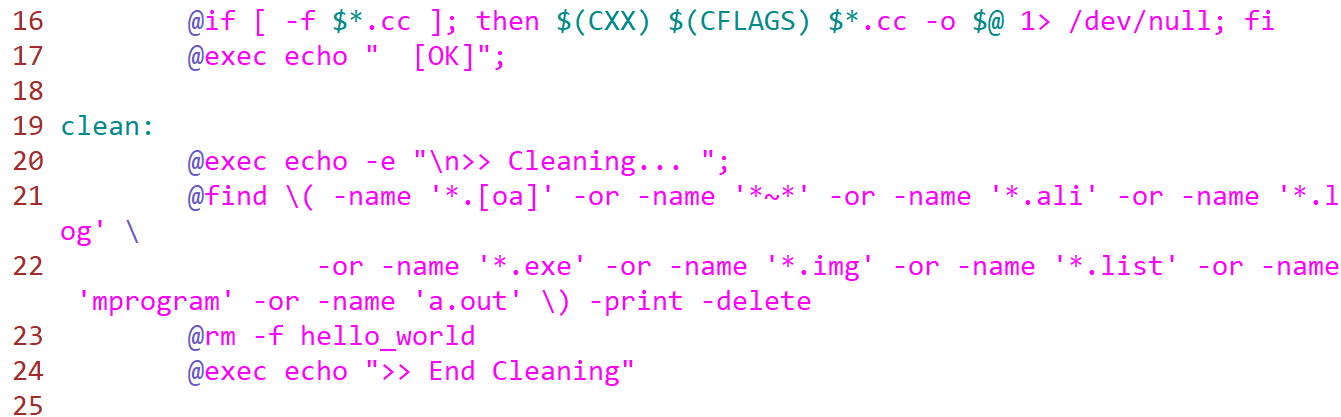
Archivo hilos.c

Construya el ejecutable correspondiente al programa anterior, primero para obtener un ejecutable sobre GNU-Linux (por ejemplo, en una máquina virtual) con el comando



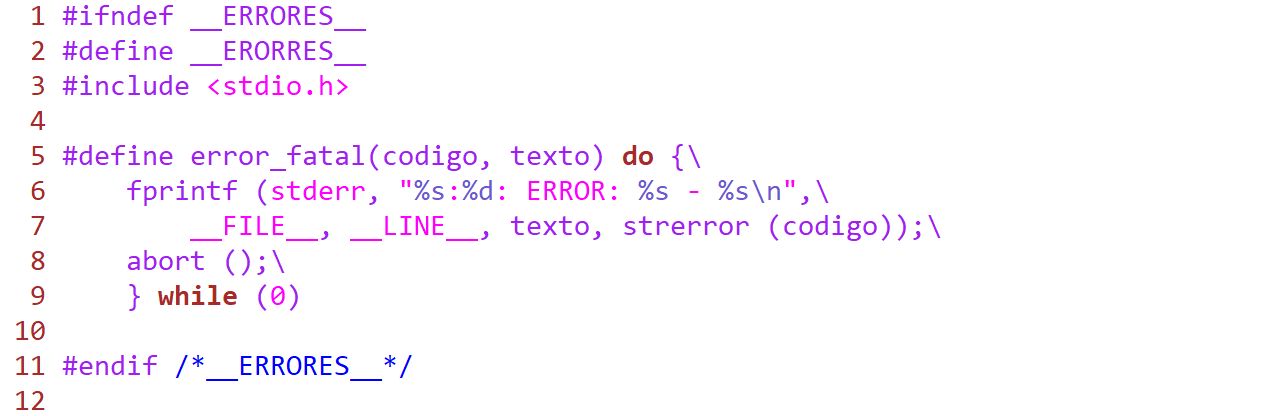
y después, obtener un ejecutable para correrlo sobre MARTE OS, copie el archivo hilos.c en un subdirectorio dentro del directorio examples de la distribución de código fuente de MaRTE OS. Para construir el ejecutable sobre MaRTE OS, utilice el siguiente archivo make.



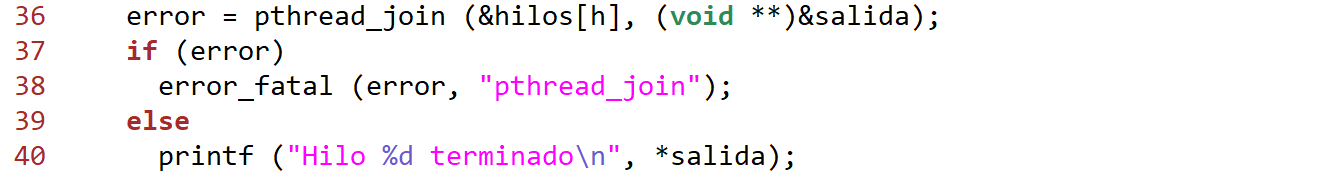


# Tratamiento de errores

En un archivo de cabecera (errores.h) coloque la siguiente macro



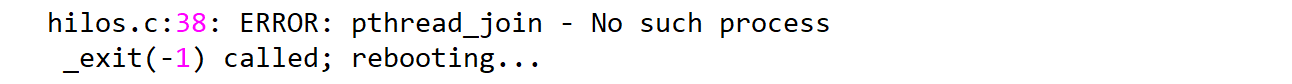
Si, por ejemplo, recodificamos el programa hilos.c e introducimos un error en la llamada a **pthread\_join**:



En ambas plataformas (GNU-Linux y MaRTE OS), se recibirán avisos al compilar el programa y al ejecutar el programa. Sobre GNU-Linux, en tiempo de ejecución se espera observar algo similar a:



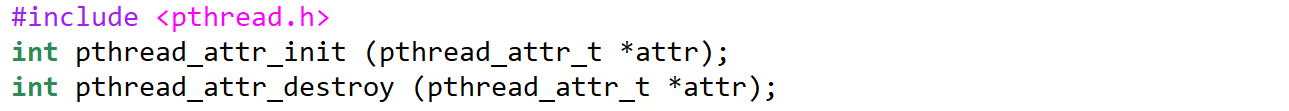
Y sobre MaRTE OS, en tiempo de ejecución algo similar a



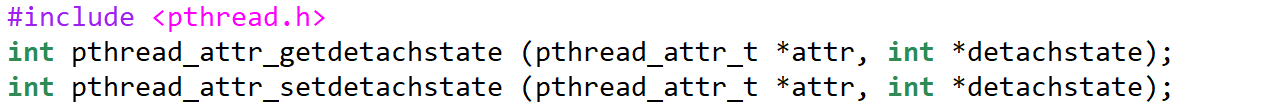
# Atributos de un hilo

En la biblioteca pthreads los hilos son considerados objetos con una implementación oculta para el usuario, por lo tanto, la definición y modificación de los atributos de estos objetos es posible únicamente a través de las funcione definidas para tal efecto. POSIX define los siguientes atributos para los hilos: vinculación –detachstate--, tamaño de la pila –stacksize--, dirección de la pila –stackaddr--, ámbito de contienda –scope--, herencia de la planificación –inheritsched--, política de distribución –schedpolicy-- y parámetros de planificación –schedparam--.

Los atributos de un hilo se almacenan en objetos del tipo **pthread\_attr\_t**, son inicializados con los valores por defecto llamando a la función **pthread\_attr\_init** y borrados con **pthread\_attr\_destroy**:



La condición de separable o no separable de un hilo se modifica con la función **pthread\_attr\_setdetachstate** y es consultada con **pthread\_attr\_getdetachstate**:



REF.

[Marquez] Francisco M. Márquez, UNIX Programación avanzada, Alfaomega, 3/a edición, 2009.