

# PROYECTO DE INVESTIGACIÓN:

## HILOS

VALENTINA BOTERO VIVAS

DOCENTE

AUGUSTO SALAZAR JIMENEZ

UNIVERSIDAD DE ANTIOQUIA

INFORMÁTICA II

15 DE JULIO DEL 2020

## Hilos

Los hilos o subprocesos son el flujo de control de datos de un programa. Es un medio que permite administrar las tareas de un procesador y de sus diferentes núcleos de una forma más eficiente.

Gracias a los hilos, las unidades mínimas de asignación, que son las tareas o procesos de un programa, pueden dividirse en trozos para así optimizar los tiempos de espera de cada instrucción en la cola del proceso.

Ahora cada hilo contiene un trozo de la tarea a realizar, algo más simple que si introducimos la tarea completa en el núcleo físico. De esta forma el sistema es capaz de procesar varias tareas al mismo tiempo y de forma simultánea, de hecho, podrá hacer tantas tareas como hilos tenga, y normalmente son una o dos por cada núcleo.

Aunque un núcleo solo puede realizar una tarea a la vez, se pueden usar los hilos para hacer creer al usuario (y al ordenador) que sí se puede hacer más de una tarea al mismo tiempo.

En lugar de realizar una tarea completa, se divide en porciones ( cada hilo se encarga de un aspecto concreto del programa ), de modo que se van alternando entre porciones de distintas tareas para que parezca que ambas se están ejecutando simultáneamente.

## Historia

En un sistema multiprogramado, los procesos constituyen una forma de expresar concurrencia, es decir, la capacidad de diferentes partes para ejecutarse fuera de orden o en orden parcial, sin afectar el resultado final. Sin embargo, hace más de medio siglo, el concepto de proceso no fue desarrollado para permitir la convivencia de los programas de distintos usuarios sobre los carísimos computadores de la época. Estos programas también se ejecutaban en concurrencia, pero solo interaccionaban entre ellos en su competencia por el acceso a los recursos compartidos. Sin embargo, enseguida se vio el potencial del modelo de procesos para implementar programas donde los procesos cooperaban, más que competían, en el acceso a los recursos.

Hacia el último cuarto del siglo XX, la informática se abrió a nuevos tipos de aplicaciones, como por ejemplo la edición de textos, pronto se vio que el modelo de procesos resultaba demasiado rígido, en particular porque los procesos no compartían memoria.

Un editor de textos debe realizar varias tareas concurrentemente en estrecha relación: lectura, corrección ortográfica, copias de seguridad... todas ellas accediendo a un mismo espacio común. Compartir ese espacio es más ágil que andar comunicando explícitamente datos de un proceso a otro.

Cuando esta necesidad se hizo evidente, los sistemas operativos introdujeron

la posibilidad de compartir segmentos de memoria entre los procesos, donde se ubicaban las variables compartidas.

Los hilos fueron una alternativa de programación más ágil, que incluía entidades de ejecución con un contexto propio, compartían las variables del proceso y podían comunicarse entre ellos de forma natural.

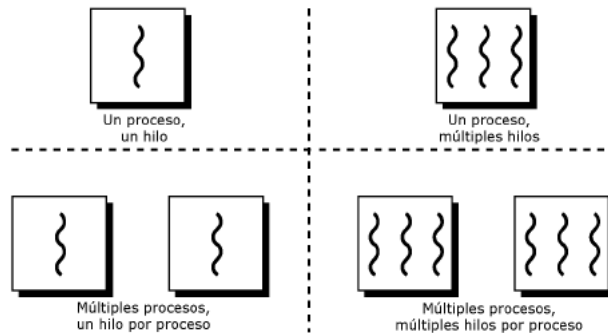


Figura 1: Hilos y procesos. Tomado de [2]

## Problemas de seguridad:

Se pueden producir dos vulnerabilidades trágicas:

- Bloqueo mutuo: Es el bloqueo irreversible de un conjunto de hilos o procesos.
- Condición de carrera: Es aquel que se produce cuando varios hilos o procesos intentan modificar de forma simultánea a un mismo recurso.

## Implementación:

Generalmente se implementan hilos de dos maneras:

- Multihilo apropiativo: permite al sistema determinar cuándo debe haber un cambio de contexto. La desventaja de esto es que el sistema puede hacer un cambio de contexto en un momento inadecuado, causando problemas.
- Multihilo cooperativo: depende del mismo hilo abandonar el control cuando llega a un punto de detención, lo cual puede traer problemas cuando el hilo espera la disponibilidad de un recurso.

Ahora, a nivel práctico los hilos pueden ser implementados a nivel de usuario o a nivel de kernel.

## Hilos a nivel de Kernel(KLT):

Todo el trabajo de gestión de hilos lo realiza el kernel o núcleo(Hardware).

Tienen como gran beneficio poder aprovechar mejor las arquitecturas de los multiprocesadores, y que proporcionan un mejor tiempo de respuesta, ya que si un hilo se bloquea, los otros pueden seguir ejecutando.

## Hilos a nivel de usuario (ULT)

Todo el manejo de los hilos lo realiza la aplicación (software) y el núcleo no es consciente de los hilos.

Tienen como beneficio que su cambio de contexto es más sencillo. Se pueden ejecutar en cualquier sistema operativo. La biblioteca de hilos es un conjunto compartido.

Los hilos tienen una desventaja y es que están limitados por las capacidades del sistema donde se corren. Es decir, la respuesta depende del hardware.

## Implementación C++

Guiada por: [5]

- Primero debemos incluir la librería Qthread la cual es un identificador de un hilo de plataforma, permite administrar el hilo monitoreando su vida útil y solicitando que finalice su trabajo.
- Se hereda la librería y redefinimos el método run() que es el que contiene el código del hilo. En este ejemplo actualizará un valor que va desde 0 a 99. Además en el constructor se recibe un parámetro que define cuanto tiempo esperará cada componente para cambiar el valor.
- Luego se debe instanciar para poder ejecutarlo. En este ejemplo se instancia 3 veces, ya que se utilizarán 3 hilos para cambiar los valores de una barra de progreso, un dial y un lcdnumber en la interfaz.
- Posteriormente se hace la conexión entre los hilos y los elementos de la interfaz.
- Para finalizar se agregan dos botones a la interfaz, los cuales darán inicio y final a la ejecución de los hilos con el comando start() y terminate() respectivamente.

En el repositorio se encuentra el video de funcionamiento y además el código.

# Bibliografía

- [1] Qué son los hilos de un procesador.Diferencias con los núcleos.(2019).Profesional Review.  
<https://www.profesionalreview.com/2019/04/03/que-son-los-hilos-de-un-procesador/Que-son-los-hilos-de-procesamiento-o-threads>
- [2] Programación con hilos.(2014). Sebastian Castro.  
<https://prezi.com/zligxav4or3z/programacion-con-hilos/>
- [3] Procesos e hilos(s.f). Universidad del país Vasco.<https://ocw.ehu.eus/pluginfile.php/12388/mod-resource/content/13/html/Recursos/P04/Procesos-y-threads.html>
- [4] Procesos Cooperativos, Hilos y Planificación de CPU(2013). Sistemas Operativos. <https://chsos20132912059.wordpress.com/2013/10/15/procesos-cooperativos/>
- [5] Tutorial Qt Creator(2015). DuarteCorporations.<https://www.youtube.com/watch?v=iWFbjTlHrKI>