

Tareas

Antes de añadir hilos:

- **¿Qué se puede notar con respecto al tiempo de ejecución? ¿Es predecible?**

El tiempo de ejecución total varía significativamente cada vez que se ejecuta el programa. No es predecible ya que esta velocidad puede variar por distintos factores, como el CPU, la velocidad de procesamiento y si hay otros procesos ejecutándose al mismo tiempo dentro del sistema operativo.

- **Nombrar un proceso o función de la vida real que pueden ser considerados procesos de “máxima velocidad posible” que dependen casi exclusivamente de la velocidad de la máquina que los ejecuta.**

Un ejemplo de este tipo de proceso es el ordenado de una lista. Independientemente del algoritmo que se use, el tiempo que se usa para ordenar una lista depende casi totalmente de la velocidad de la CPU. Cuanto más rápido sea el procesador, más rápido se completará el proceso para ordenar.

- **Nombrar un proceso o función de la vida real que pueden ser considerados procesos de “velocidad de respuesta no dependiente de la velocidad de procesamiento” o que sea de naturaleza impredecible o externa.**

Un ejemplo de proceso impredecible puede ser la comunicación de red con un servidor remoto. El tiempo que tarda un servidor en recibir una respuesta de un cliente puede depender de varios factores como la velocidad de la conexión, su calidad, la distancia física, entre otros. Todos estos factores operan por fuera de la velocidad de procesamiento y hacen que sea prácticamente imposible predecir la velocidad de ejecución.

Después de añadir hilos:

- **¿Qué se puede notar con respecto al tiempo de ejecución? ¿Se mejoró el tiempo de respuesta con respecto al mismo programa sin hilos?**

Después de añadir hilos, el tiempo de ejecución se redujo ligeramente en cada tarea, dando como resultado una reducción en el tiempo total. El código es más eficiente y las tareas se completan al mismo tiempo.

- **¿Compleatan las funciones su ejecución en el orden establecido?**

Las funciones se completan en el orden establecido, pero en computadoras con velocidades distintas podría no ser el caso.

- **Nombrar un escenario real donde el multi-hilado puede mejorar considerablemente el tiempo**

de respuesta de un sistema.

El multihilado podría ayudar al cargar una página de internet, ya que la descarga y procesamiento de los recursos se ejecutaría en varias tareas al mismo tiempo y darían una carga mas fluida al usuario.

Sumador/Restador

Despues de añadir RaceCondition:

- **¿Qué se puede notar con respecto al tiempo de ejecución?**

El tiempo de ejecución es practicamente igual, no hay una diferencia clara entre la versión base y la versión con RaceCondition.

- **¿Qué sucede con el valor final del acumulador?**

El valor final se mantiene igual que antes de añadir la RaceCondition, devolviendo el resultado "0" al final del programa.

- **¿Por qué sucede esto?**

Los hilos no estan sincronizados y modifican la variable "acumulador" indefinidamente, por lo que un hilo puede leer su valor mientras otro esta actualizandolo. Es por eso que el resultado final no es el correcto.

- **¿Cómo se puede corregir esta condición de carrera sin dejar de utilizar hilos?**

Se puede utilizar algún mecanismo de sincronización como un semáforo o un lock. Esto aseguraria que solo un hilo pueda modificar "acumulador" a la vez, evitando así que los hilos se solapen.

Luego de corregir:

- **¿Qué sucede con el valor final del acumulador?**

Despues de ejecutar el programa, el valor es "5000", funcionando correctamente.

- **¿Qué se puede notar con respecto al tiempo de ejecución?**

El tiempo de ejecución es ligeramente mas extenso ya que cada hilo debe adquirir y liberarse en cada ejecución. Este aumento es apenas notable y se puede ignorar cuando el programa funciona correctamente luego de añadirlo.