Arquitectura de Sistemas Operativos.

TUP - 2024

Trabajo Practico nro. 4.

Hilos:

1. Tarea sin hilos.
   1. Al ejecutar varias veces el script, puedo notar que el tiempo de ejecución podría ser considerablemente predecible. En mi caso particular, durante todas las ejecuciones, el tiempo de ejecución total estuvo en el rango de los 6.3 segundos. Pero este resultado dependerá del poder de procesamiento de la máquina que ejecute el script, ya que, aunque las tareas dos y tres tienen un tiempo de ejecución fijo en un caso óptimo, esto puede no ser así y tardar más de lo que está definido.
   2. Un ejemplo de la vida real de un proceso que pueda ser considerado de “máxima velocidad posible” podría ser el programa de lavado rápido de un lavarropas automático, ya que la rapidez de este proceso dependerá casi únicamente de la optimización de este proceso en el lavarropas.
   3. Un proceso de la vida real que pueda ser considerado como “velocidad de respuesta no dependiente de la velocidad de procesamiento” podría ser preparar un asado, ya que este proceso dependerá de quien prepare el asado y de los gustos o costumbres del mismo.
2. Tarea con hilos
   1. En mi caso, la ejecución del programa con hilos tuvo un cambio significativo o notable. El tiempo de respuesta o finalización del script fue, en promedio, de 4 segundos.
   2. En todas las ejecuciones que realicé, las tareas se ejecutaron siempre en el mismo orden (1, 2, 3), pero finalizaron en orden distinto (ej. 2, 1, 3).
   3. Un ejemplo de uso práctico de multihilos podría ser el proceso de recorrer una lista. Utilizando un proceso que paralelice los pasos, logramos que el tiempo de la acción que se quiera realizar (ej. buscar un elemento en la lista) sea mucho inferior y el proceso sea mucho más óptimo.
3. Tarea con race
   1. El tiempo de ejecución, en mi caso, está en el orden de 0,12 segundos. Este tiempo dependerá totalmente del procesador.
   2. El valor final de la variable acumulador oscila entre los valores 5000 y -5000.
   3. Este problema sucede debido a la condición de carrera, es decir, que los hilos acceden y modifican una variable compartida sin una sincronización adecuada. Específicamente, en el script, tanto el hilo “sumador” como el hilo “restador” acceden y modifican la variable acumulador al mismo tiempo sin ningún mecanismo de protección, y por esto el resultado es inconsistente.
   4. Para solucionarlo deberíamos implementar algún mecanismo de control entre los hilos, que impida este comportamiento, para lograr el funcionamiento deseado.
4. Sin race
   1. El valor final de la variable acumulador es siempre 0.
   2. b. El tiempo de ejecución parece ser consistente con el tiempo de ejecución del script anterior.