

Sistemas Empotrados y de Tiempo Real:

Práctica 1: Multithreading en C y GNU/Linux (REENTREGA)

1. Ejecutar los siguientes casos y justificar su comportamiento:

a. `./practica1 (multicore)`

Al ejecutar este caso los valores del coste varían en el rango 0.5-0.6, ya que al ser multicore cada hilo se ejecuta en una CPU diferente y por ello no hay fallos temporales.

b. `./practica1 (monocore)`

Al ejecutar este caso los valores del coste varían en el rango 2.05-2.11 (todos fallos temporales), ya que al ser monocore todos los hilos se ejecutan en la misma CPU y deben turnarse para realizar las tareas.

c. `./practica1 (monocore) + stress`

Al ejecutar este caso los valores del coste varían en el rango 2.10-2.16 (todos fallos temporales), ya que al ser monocore todos los hilos se ejecutan en la misma CPU y deben turnarse para realizar las tareas, y al estresar el sistema esto hace que les cueste más a los hilos turnarse para realizar las tareas.

d. `./practica1 (multicore) + stress`

Al ejecutar este caso los valores del coste varían en el rango 0.56-0.86, ya que al ser multicore cada hilo se ejecuta en una CPU diferente y por ello no hay fallos temporales (en algún caso que he ejecutado de este tipo puede aparecer algún que otro fallo temporal, pero en la mayoría de los casos se cumplen las restricciones temporales con un coste algo más alto que en el caso en el que no estresamos el sistema).

2. ¿En qué casos de ejecución (nombrados anteriormente) el sistema es capaz de cumplir las restricciones temporales (tanto tiempo de cómputo como periodicidad)?

En el multicore.

3. ¿Qué número mínimo de CPUs se necesitan para que tu programa ejecute correctamente sin fallos de restricciones temporales? Usa el comando `taskset` para comprobarlo.

Sin estresar el sistema y ejecutando el comando `taskset` para 3 CPUs, logramos que se cumplan las restricciones temporales en todos los casos, obteniendo los valores del coste en el rango 0.52-0.66.

Estresando el sistema y ejecutando el comando `taskset` para 4 o más CPUs, logramos que se cumplan las restricciones temporales en todos los casos (salvo en alguno que pueda aparecer algún que otro fallo temporal), obteniendo los valores del coste en el rango 0.65-0.85.

4. ¿Qué solución se podría proponer para cumplir plazos estrictos temporales de periodicidad en la ejecución de los threads utilizando la configuración actual que tienen los ordenadores del laboratorio?

Ampliar el periodo para que pueda acabar de ejecutar la orden en el monocore antes de que acabe el deadline.