

1. Ejecutar los siguientes casos y justificar su comportamiento:

**A. ./practica1 (multicore)**

En este caso, en la mayoría de los casos el tiempo usado no sobrepasa los 0.6 segundos, salvo ciertos casos aislados, esto tiene sentido pues el ordenador puede aprovechar sus 4 núcleos para ejecutar los distintos hilos.

**B. ./practica1 (monocore)**

En este caso, podemos ver como ninguno de los hilos cumple con los deadtimes, esto se debe a que un solo núcleo debe lidiar con 4 hilos a la vez lo que hace que el tiempo por hilo se acerque a los 2 segundos.

**C. ./practica1 (monocore) + stress**

En este caso, podemos ver que ocurre lo mismo que en el caso de monocore, pero los tiempos son más altos, puesto que la cpu esta estresada.

**D. ./practica1 (multicore) + stress**

En este caso, podemos ver que los tiempos se vuelven más dispersos, lo que hace que, aunque en su mayoría cumpla el deadline, algunos consigan superarlos, esto tiene sentido, puesto que las cpus están estresadas, por lo que habrá veces que los hilos tengan que esperar para entrar y en raras ocasiones no tengan que esperar, o no tanto por lo menos.

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 caso de multicore, y en algunos casos del multicore + stress.

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.

Se necesitan 2 nucleos para que ejecute correctamente sin fallos de restricciones temporales

4. ¿Qué solución se podría proponer para cumplir plazos estrictos temporales de

periodicidad en la ejecución de los threads SIN cambiar la configuración actual que tienen los ordenadores del laboratorio?

Puesto que no se puede cambiar la configuración actual de los ordenadores del laboratorio, la única solución para que cumpla con plazos estrictos temporales de periodicidad pasa por tratar de reducir el coste de ejecución de los threads, o aumentando el plazo temporal.