

Programación Concurrente y de Tiempo Real

PRÁCTICA 2

M^a Teresa Supervielle Sánchez

1. Ejercicio 1: hebra.java y Usa_hebra.java

n	1ª ejecución	2ª ejecución	3ª ejecución	4ª ejecución	5ª ejecución
10	0	0	0	0	0
100	0	0	0	0	0
1000	0	0	0	-231	0
10000	-4220	1128	-1102	-1496	-3876
100000	-2659	-12594	-2300	-9969	-9123
1000000	-867392	-6616	369768	723815	96984

Cuadro 1: Resultados hebra.java

Como hemos estudiado en la asignatura, una condición de concurso se dan cuando dos o más tareas concurrentes acceden a recursos compartidos entre ellas al mismo tiempo sin control de exclusión mutua.

En el caso de este programa la condición de concurso se da cuando se crean dos hilos que acceden a operaciones contrarias, como un incremento y un decremento.

Si tenemos dos hilos que van accediendo intercaladamente a las operaciones dichas anteriormente el resultado debería ser 0, como es el caso en las n de menor tamaño. Por otro lado, cuanto mayor sea el número de iteraciones a realizar, este acceso a una única memoria puede verse sobrescrito por el descontrol provocado en los hilos, dando números muy dispares a 0.

2. Ejercicio 2: tareaRunnable.java y Usa_tareaRunnable.java

n	1ª ejecución	2ª ejecución	3ª ejecución	4ª ejecución	5ª ejecución
10	0	0	0	0	0
100	0	0	0	0	0
1000	108	0	-92	0	-138
10000	-12	-2795	-453	1629	-2188
100000	-24736	1522	-6250	10738	364 94
1000000	20411	-6363	12708	12910	-3097

Cuadro 2: Resultados tareaRunnable.java