

# PCTR - Práctica 4

Javier López Sierra

Noviembre 2022

## 1. Ejercicio 1

Al ejecutar *tryThree.java* solo observamos que no se termina la ejecución por mucho tiempo que esperemos. Puedo asumir que evidentemente se trata del problema de espera infinita asociado a este algoritmo, en el cual ambos hilos se han quedado atrapados en el bucle de espera, esperando cada uno a que el otro hilo termine, sin poder terminar ninguno de los dos.

Al ejecutar la versión *tryFour.java* se comprueba que en este caso sí funciona, y da el resultado esperado,  $n=0$ . Es evidente que haber añadido operaciones para ocupar tiempo entre la cesión y solicitud de turno es efectivo, aunque resulte poco eficiente.

## 2. Ejercicio 2

Al probar el algoritmo de *Dekker* podemos comprobar que es efectivo al controlar el acceso de ambos hilos a la variable  $n$  (el resultado es  $n=0$ ), aunque para el caso concreto que hemos implementado, no podemos comprobar si existe una mejora en la eficiencia con respecto al algoritmo implementado en el ejercicio anterior, por la simplicidad de las operaciones.

## 3. Ejercicio 3

Al implementar el algoritmo de *Peterson* podemos observar que funciona y es efectivo como el algoritmo de *Dekker*. A título personal, la implementación de este algoritmo la considero visualmente más fácil de entender y más sencilla de implementar.