

# Práctica 4 - Ejercicios 1, 2, 3

federico.carrilloch@alum.uca.es

August 2020

## 1 Algoritmo de Dekker

Con este algoritmo solucionamos el problema de la región crítica mediante el uso de una variable turno (global a todos los procesos y se alterna) y de un array con tantos elementos como procesos existan (usamos el **tipo BOOLEAN**) para indicar la intención del proceso de tomar el recurso compartido (true) o que está siendo usado por otro (false). La bandera se recorre mediante un índice que se va igualando a un identificador único para cada hilo, el cual es asignado cuando llamamos al constructor de cada hilo.

## 2 Algoritmo de Eisenberg y McGuire

El primer software que logró solucionar el problema de acceso a la sección crítica llegó implementado este algoritmo. Se basa nuevamente en un sistema de turnos y banderas (pero cambiando que al inicio de la ejecución el turno es asignado de manera aleatoria entre 0 y N, siendo  $N = n^{\circ}\text{hilos} - 1$ ). En este caso como banderas usamos un array de enum con tres posibles estados:

- **IDLE** (Ocioso, se trata del estado inicial de cada proceso; significa que ni está en espera ni está usando el recurso),
- **WAITING** (En espera para entrar en la sección crítica, sirve para que cada proceso anuncie que requiere del recurso compartido)
- **ACTIVE** (Cuando el proceso se apropie de la sección crítica).

## 3 Algoritmo de la panadería de Lamport

Este algoritmo emplea los métodos **LOCK** y **UNLOCK**. El método **LOCK** bloquea el hilo mediante un sistema de tickets y turnos. Una vez que el hilo sale de **LOCK** accede a la sección crítica y en el método **UNLOCK** su turno se vuelve a establecer a cero para a continuación volver a ser bloqueado (en nuestro caso). El nombre de dicho algoritmo viene por el sistema que se usaban en las panaderías, según Wikipedia, en las cuales se daría un ticket a cada cliente que

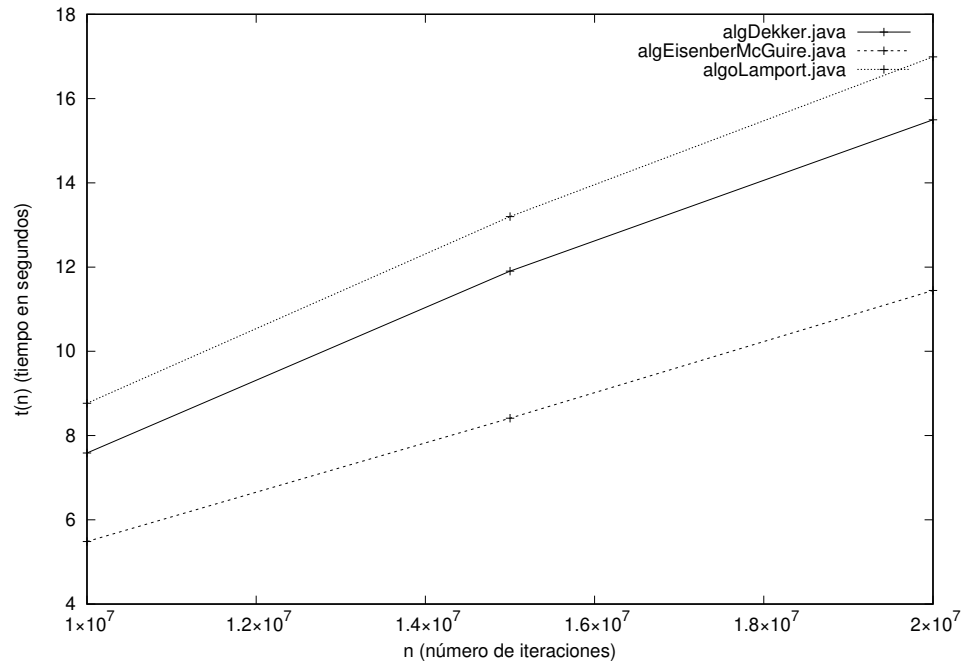


Figure 1: Gráfica con los tiempos de cada algoritmo usando 2 hebras.

iría accediendo al recurso compartido (para comprar pan) cuando le tocara. Si el cliente quiere comprar más se debe situar al final de la cola para avanzar de nuevo.

*En este algoritmo es con el único que no hemos tenido un resultado esperado, es decir, había fallos en la modificación de la variable compartida y distaba el resultado (en unas 5 o 6 unidades) del valor esperado de manera teórica.*

## 4 Gráfica