

**Escuela Técnica Superior de Ingeniería**

---

# Prácticas de Programación Concurrente y Distribuida

---

**3º Curso de Grado en Ingeniería Informática**

## **PRÁCTICA 5**

Condición de Sincronización en JAVA

El objetivo de la práctica es familiarizarse con los mecanismos de Java para la sincronización de hilos. Para ello, se resolverá el siguiente supuesto:

*En una estación de lavado de coches hay tres túneles de lavado dispuestos de forma paralela. A dicha estación llegan para lavarse turismos y furgonetas. Cualquiera de los vehículos podrá usar cualquiera de los túneles, pero si hay una furgoneta en el túnel central no podrá haber otra furgoneta en ninguno de los túneles laterales.*

El proyecto para solucionar el enunciado contendrá las siguientes clases:

- **Coche.** Representará, mediante un hilo, cada uno de los coches que llegan al túnel de lavado. El hilo se creará heredando de la clase `Thread`. El tiempo de lavado será aleatorio entre 1 y 3 segundos.
- **Furgo.** Representará, mediante un hilo, cada uno de las furgonetas que llegan al túnel de lavado. El hilo se creará implementando el *interface* `Runnable`. El tiempo de lavado será aleatorio entre 1 y 3 segundos.

- **Generador.** Contendrá el método `main` y será quién comience la ejecución. Debe lanzar 10 vehículos a intervalos de tiempo de entre 1 a 2 segundos. El tipo de vehículo generado será aleatorio. La ejecución finalizará cuando todos los hilos hayan finalizado.
- **Tunel.** La clase `Tunel` controlará la entrada de los vehículos e implementará los siguientes métodos:
  - **EntraCoche.** Que deberá ser invocado por los coches que quieren entrar al túnel.
  - **SaleCoche.** Que deberá ser invocado por los coches cuando salen del túnel.
  - **EntraFurgo.** Que deberá ser invocado por las furgonetas que quieren entrar al túnel.
  - **SaleFurgo.** Que deberá ser invocado por las furgonetas cuando salen del túnel.

El control de la concurrencia y la sincronización se realizará en la clase `Tunel`, mediante las primitivas de Java `wait()`, `notify()` y/o `notifyAll()`

Además, se deberá crear un *Frame* que visualice de forma gráfica, mediante un *Canvas*, la situación de los túneles y las colas de espera. El nivel de representación gráfica es libre, pero al menos, se deberán identificar los dos tipos de vehículo, la ocupación del túnel y las colas de espera.