

CÓDIGO **SINCRONIZADO Y** **USO DE VALORES** **ATÓMICOS**



INTERFERENCIA ENTRE HILOS

- ▶ ¿Qué sucede si dos hilos acceden a la misma variable a la vez, alguno de ellos en **modo escritura**?
- ▶ Una sentencia java es, a bajo nivel, varias instrucciones del procesador.
- ▶ Condición de carrera: *no podemos **asumir** el orden en el que el procesador va a ejecutar las instrucciones de dos hilos diferentes.*

SYNCHRONIZED

- ▶ Métodos o bloques.
- ▶ Indican que ese bloque de código (o método) puede ser ejecutado solo por un hilo a la vez. El otro debe esperar.

```
public synchronized void incrementar() {  
    c++;  
}
```

```
synchronized(this) {  
    count++;  
}
```

CERROJOS INTRÍNSECOS O MONITOR

- ▶ Entidad interna que permite la sincronización.
- ▶ Cada objeto tiene su propio cerrojo intrínseco.
- ▶ Un método sincronizado adquiere el cerrojo de un objeto, y lo libera cuando termina.
- ▶ Si un hilo tiene el cerrojo de un objeto compartido, otro no puede adquirirlo hasta que el primero lo libera.

VARIABLES ATÓMICAS

- ▶ En el paquete *java.util.concurrent.atomic*
- ▶ Tipos básicos que ya están sincronizados (*thread-safe*).
- ▶ Métodos *get*, *set*, *counterAndSet*, ...
- ▶ *AtomicBoolean*, *AtomicInteger*, *AtomicLong*, *LongAdder*, *DoubleAdder*, ...

<https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/atomic/package-summary.html>