

# Programmazione Concorrente e Parallela - Serie 4

Maura Clerici

20.03.2019

## Esercizio 2

La classe Sensore dispone di un metodo booleano `resetIfAbove()` che permette di capire se la soglia di quel determinato sensore è stata superata. All'interno del metodo `run()`, infatti, il sensore continua a verificare se la soglia è stata superata grazie al valore `true` o `false` ritornato dal metodo.

In `resetIfAbove()`, per introdurre l'utilizzo di `compareAndSet()` sulla variabile atomica del contatore, sono state definite delle variabili intere per il valore precedente ed il valore attuale del contatore. Se il valore del contatore è più basso della soglia, il metodo ritorna `false`, in modo da poter continuare a verificare quando la soglia non viene superata. Altrimenti, se il valore del contatore è uguale o più alto della soglia, lo stesso viene assegnato al valore precedente della soglia ed il nuovo valore viene impostato a 0. I due valori vengono poi passati al metodo `compareAndSet()` che si occupa di assegnare il nuovo valore al contatore azzerandolo e di ritornare `true`, in modo da permettere al sensore di uscire dal ciclo di controllo e di fermarsi.

```
private boolean resetIfAbove() {
    int valorePrecedente, valoreAttuale;
    int currentAmount = S4Esercizio2.counter.get();
    if (currentAmount < soglia) {
        return false;
    } else {
        valorePrecedente = S4Esercizio2.counter.get();
        valoreAttuale = 0;
        return S4Esercizio2.counter.compareAndSet(valorePrecedente, valoreAttuale);
    }
}
```

## Esercizio 4

Per utilizzare degli oggetti immutabile, utilizzati come risorse nella classe `Coordinate`, le variabili `lat` e `lon` sono state definite come `final`:

```
class Coordinate {  
    private final double lat;  
    private final double lon;
```

È stato creato un costruttore per dare la possibilità di definire i loro valori al momento della creazione delle istanze, valori che non potranno più essere modificati:

```
    public Coordinate(final double mlat, final double mlon) {  
        lat = mlat;  
        lon = mlon;  
    }
```

In questo modo, le due variabili vengono definite attraverso il costruttore al momento della creazione:

```
    public void run() {  
        while (!S4Esercizio4.completed) {  
            S4Esercizio4.curLocation = new Coordinate(ThreadLocalRandom.current().nextDouble(), ThreadLocalRandom.current().nextDouble());  
            // Wait before updating position  
            try {  
                Thread.sleep(ThreadLocalRandom.current().nextLong(1, 5));  
            } catch (final InterruptedException e) {  
                Thread.currentThread().interrupt();  
            }  
        }  
    }
```