

# Programmazione Orientata agli Oggetti

## 28 luglio 2022

### Cosa fare

Si progetti una o più gerarchie di tipo in Java in modo da supportare le operazioni indicate nel tema specificato sotto. Definire **classi e interfacce**, le signature dei **metodi**, e gli **stati astratti e concreti**, specificando il **ruolo** e **protocollo** delle classi/interfacce e i **contratti** dei metodi più importanti; **implementare lo stato concreto e i metodi**. Valuterò l'elaborato in base alla qualità del progetto e della sua implementazione (responsabilità, tipologie, contratti, ADT, parametrizzazione, qualità del body, pattern di design, astrazioni, incapsulamento). Non leggerò spiegazioni complesse, schemi e diagrammi.

Non consegnare il compito se non si sono implementati almeno i metodi per le operazioni principali e un Main che funga da client per questa API con esempi di chiamate di ciascuna delle operazioni principali.

La soluzione va scritta su dei file di testo – in particolare non binari - (ad es. usando IntelliJ, Eclipse o altro a piacere) che devono essere caricati sul sistema elearning.uniud.it entro il termine di scadenza (non ci sono deroghe se non per casi particolari concordati a priori e giustificati adeguatamente).

Scrivere i propri nome e matricola su ciascuno dei file con la soluzione.

## Sensori

L'azienda E-barche realizza e vende soluzioni di strumentazione elettronica e informatica per barche a vela che consente all'equipaggio di avere sott'occhio tutti i dati relativi allo stato della barca, della navigazione e poter agire su qualche dispositivo. In particolare, gli strumenti consentono di visualizzare la posizione geografica data dal GPS (latitudine e longitudine, come ad es. 45.3456 gradi N e 13.5432 gradi E), di proiettarla su una mappa geografica/nautica, di visualizzare la direzione di movimento della barca (fornita dalla bussola, in gradi bussola: 0 per il Nord, fino a 359), velocità di movimento (fornita dal solcometro), velocità e direzione del vento, inclinazione longitudinale (beccheggio, in gradi rispetto all'orizzontale: es. -5 gradi ~ inclinata verso poppa di 5 gradi) e laterale (rollio, in gradi rispetto all'orizzontale: es. -5 gradi ~ inclinata 5 gradi a sinistra), profondità del mare (in metri), distanza e direzione bussola rispetto a un eventuale punto di riferimento (come ad es. il porto di Trieste). Il timone consente di sapere che angolo ha rispetto all'asse longitudinale della barca (es. -5 gradi ~ pala del timone orientata 5 gradi a sx). Esiste anche l'autopilota che, se attivato su un certo valore di gradi bussola, mantiene quella direzione di moto della barca. E il motore che, se acceso, funziona a marcia avanti o indietro con un certo numero di giri; valori che possono essere interrogati.

Ogni dispositivo (scandaglio, anemometro, accelerometri per l'inclinazione, GPS, bussola, solcometro) espone degli endpoint che consentono di "leggere" i valori, che vengono aggiornati automaticamente con una certa frequenza (ad es. 5Hz, 5x al secondo).

Serve progettare e realizzare una API che consenta di periodicamente interrogare ciascun dispositivo, raccogliere i valori aggiornati e metterli a disposizione ai client dell'API, sia per quanto riguarda i valori "istantanei" che delle medie mobili (ad es. la media della velocità negli ultimi 2 secondi), e fornisce endpoint per accendere/spegnere il motore, accelerare o rallentare, andare avanti o indietro; e endpoint per attivare/disattivare l'autopilota su una certa direzione.

Progettare e implementare l'API almeno per queste operazioni:

- \* **extractValue(device, ....)** che espone il valore attuale di una certa grandezza gestita dal dispositivo (es. latitudine del GPS)
- \* **movingAverage(device, ....., N\_seconds)** per calcolare la media mobile di una certa grandezza
- \* **sendCommand(device, cmd, ...)** per spedire un comando (es "accenditi", "acceletera a 2000 giri") a un dispositivo
- \* **configureDevice(...)** che consente di aggiungere, o rimuovere, un dispositivo dalla barca. Se aggiunto può essere manipolato come sopra, se tolto non potrà più venir usato.

**Obbligatorio:** scrivere un **main** che contenga chiamate a ciascuno di questi metodi.