

Università di Ferrara
Corso di Sistemi Operativi e Laboratorio
Prova di Java
15 Giugno 2023 – durata 1h

Si realizzi un programma Java multi-threaded per monitorare il consumo di carburante di un Boeing 747. Per misurare il consumo di carburante si supponga di utilizzare l'unità di misura litri al secondo (l/s). Ad esempio, un valore di 10 l/s indica che l'aereo ha consumato 10 litri nell'ultimo secondo.

Il thread *Main* crea il thread *SimulaConsumoCarburante* che genera una volta al secondo valori interi casuali compresi tra 0 e 25. Ad esempio, 10 significa che l'aereo ha consumato 10 litri di carburante nell'ultimo secondo. Tale valore deve essere sottratto al valore *int carburante*, che rappresenta la quantità di carburante attualmente a disposizione. Si assuma che all'avvio del thread *SimulaConsumoCarburante* il velivolo abbia un pieno di carburante di 200000 litri.

Successivamente il thread *SimulaConsumoCarburante* invia al thread *Main* un oggetto *CarburanteAttuale* tramite pipe di tipo Object. L'oggetto *CarburanteAttuale* è composto da 2 valori: *int carburante* (che indica il livello attuale di carburante in litri) e *long timestamp* (Unix timestamp del momento in cui è stata prodotta la rilevazione, vedi metodo *long System.currentTimeMillis()*).

Il thread *SimulaConsumoCarburante* deve estendere la classe Thread.

Il thread *Main* chiede all'utente (via terminale) un valore soglia di allarme espresso in litri (ad esempio 3000). Per ciascuna rilevazione ricevuta dal thread *SimulaConsumoCarburante*, il thread *Main* stampa a video l'attuale livello di carburante. Se il livello di carburante è inferiore al valore di soglia indicato, allora il thread *Main* stampa a video un messaggio di warning.

Dopo 60 rilevazioni, il thread *Main* termina il thread *SimulaConsumoCarburante* con l'opportuno metodo *termina()* e ne attende la terminazione. Attenzione, il thread *SimulaConsumoCarburante* deve terminare immediatamente.