

SAPIENZA Università di Roma  
Facoltà di Ingegneria dell'Informazione, Informatica e Statistica  
Corso di Laurea in Ingegneria Informatica ed Automatica  
**Corso di Progettazione del Software**  
Esame del **10 settembre 2019**  
*Tempo a disposizione: 3 ore*

**Requisiti.** L'applicazione da progettare riguarda un sistema ferroviario. Ogni vagone di un treno è collegato al precedente e al successivo. Il vagone di testa non ha un precedente e l'ultimo non ha un successivo. Di ogni vagone interessa il codice (una stringa). Un treno di cui interessa il nome (una stringa) contiene una serie di vagoni, di cui interessa solo il primo dal momento che i successivi sono semplicemente quelli che seguono. Non è di interesse risalire dal vagone al treno a cui appartiene, solo l'opposto. Alcuni vagoni sono trainanti, e di questi interessa il numero di CV di potenza (un intero). Altri sono vagoni letto e di questi interessa il numero di posti letto. Ogni treno ha un certo numero arbitrario di dipendenti ad esso assegnati. Di ogni dipendente interessa il nome e gli anno di servizio. Ogni treno ha un dipendente con il ruolo di capotreno.

Una parte specifica dell'applicazione riguarda il movimento dei vagoni. In particolare, un vagone può trovarsi in *sosta* o in *movimento*. La transizione dalla sosta al movimento viene comunicata con un evento *partenza*, mentre da movimento a sosta dall'evento *arrivo*. Un vagone in sosta che non sia attaccato a nessun altro può venire attaccato in coda a un altro vagone. Questo evento si chiama *sposta* con payload il vagone a cui attaccarsi e causa la verifica che quest'ultimo sia effettivamente un vagone di coda, ossia non ha già un successivo. La verifica avviene attraverso il seguente protocollo: il vagone da attaccare invia un evento *arrivo* al vagone a cui va attaccato; questo risponde *libero* oppure *occupato*. Solo nel primo caso lo spostamento avviene (con gli opportuni aggiornamenti del diagramma delle classi), nel secondo no.

Siamo interessati alla seguente attività principale. L'attività prende in input un treno  $T$  e un insieme  $V$  di vagoni e verifica che ciascun vagone non sia collegato ad altri e che sia presente almeno un vagone motorizzato. Se la verifica non va a buon fine, l'attività termina segnalando in output un errore. Altrimenti concorrentemente esegue le seguenti due sottoattività: (i) formazione e (ii) orario. La sottoattività di formazione del treno (i) forma il treno eseguendo spostamenti dei vagoni in  $V$  (i dettagli non interessano). La sottoattività chiede l'orario di orario di partenza e la tratta da percorrere e calcola l'orario di arrivo dei  $T$  (i dettagli non interessano). Quando entrambe le sottoattività sono terminate, viene stampato l'orario di arrivo del treno.

**Domanda 1.** Basandosi sui requisiti riportati sopra, effettuare l'analisi producendo lo schema concettuale in UML per l'applicazione, comprensivo di: diagramma delle classi (inclusi eventuali vincoli non esprimibili in UML); diagramma stati e transizioni per la classe *Vagone*; diagramma delle attività; specifica del diagramma stati e transizioni; segnatura dell'attività principale, sottoattività non atomiche, atomiche e segnali di input/output. Si noti che NON è richiesta la specifica delle attività. Motivare, qualora ce ne fosse bisogno, le scelte di progetto.

**Domanda 2.** Effettuare il progetto, illustrando i prodotti rilevanti di tale fase e motivando, qualora ce ne fosse bisogno, le scelte di progetto. In particolare definire SOLO le responsabilità sulle associazioni del diagramma delle classi (nella tabella, inserire anche il motivo di ognuna delle responsabilità).

**Domanda 3.** Effettuare la realizzazione, producendo un programma JAVA e motivando, qualora ce ne fosse bisogno, le scelte di progetto. In particolare realizzare in JAVA SOLO i seguenti aspetti dello schema concettuale:

- La classe *Vagone* con la classe *VagoneFired*, le classi JAVA per rappresentare le *associazioni* di cui la classe *Vagone* ha responsabilità.
- L'*attività principale* e le sue eventuali sottoattività NON atomiche.