

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 **15 febbraio 2019**
Tempo a disposizione: 3 ore

Requisiti. L'applicazione da progettare riguarda una compagnia di noleggio limousine con autista. La compagnia opera attraverso vari depositi. Di ogni deposito interessa la superficie in metri quadri gli autisti che vi operano (almeno uno) e la macchine che gestisce (almeno una). Di ogni autista è di interesse il nome e il tipo di patente (una lettera da B a F) e il deposito a cui afferisce. Delle macchine interessa la targa e se sono disponibili o in riparazione (un booleano). Alcune macchine sono lunghe e richiedono la patente D, altre hanno la struttura di pulmini e richiedono la patente C. Delle macchine lunghe viene memorizzata la lunghezza, di quelle di tipo pulmino il numero massimo di passeggeri. Un cliente, di cui interessa il numero di carta di credito, prenota corse. Le corse hanno un identificativo, un orario e sono relative a una macchina che deve essere disponibile.

Occorre seguire il comportamento degli autisti. Un autista è inizialmente a riposo. Se riceve una richiesta di operare una corsa, nel caso in cui abbia la patente idonea, si mette alla guida. Quando arriva a destinazione (segnalato da un opportuno evento) torna a riposo. Se invece non ha la patente idonea chiede ad un collega del suo deposito con la patente idonea di sostituirlo (si assuma di avere una funzione predefinita di scelta e che ci sia sempre almeno un collega con la patente idonea disponibile) e si mette in attesa della risposta da parte del sostituto. Quando questa arriva torna a riposo. Se mentre a riposo riceve una richiesta di sostituzione per una corsa da parte di un collega allora si mette alla guida notificando il collega, e come detto prima, quando arriva a destinazione torna a riposo.

Siamo interessati alla seguente attività principale che viene eseguita giornalmente. Dato un deposito, all'inizio della giornata, una attività atomica determina l'insieme delle macchine disponibili e di quelle in manutenzione. Questi due insiemi vengono usati poi nelle due sottoattività concorrenti, una di gestione delle corse e l'altra di manutenzione. La prima sottoattività fa partire tutti gli autisti e le macchine disponibili (i dettagli non interessano) e termina quando arriva il segnale di I/O di fine giornata. La seconda si occupa di riparare le macchine (i dettagli non interessano) producendo la lista delle macchine che possono tornare in servizio. Quando queste due sottoattività terminano viene stampata la lista delle macchine ancora in riparazione.

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 vincoli non esprimibili in UML); diagramma stati e transizioni per la classe *Autista*; 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.

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