

Esame Es.20230113 – Prova scritta del 13 gennaio 2023

Si vuole progettare *EasyToll*, un'applicazione per la gestione della rete autostradale nazionale. *EasyToll* deve permettere di rappresentare la struttura della rete autostradale italiana. Ogni autostrada nella rete è identificata da un codice alfanumerico univoco, da un nome (opzionale) e dalle due località ai suoi estremi (ad es., l'autostrada "A1" di nome "Autostrada del Sole" collega Milano a Napoli).

I caselli permettono l'ingresso e l'uscita dei veicoli in diversi punti della rete. Per ogni casello il sistema deve poter rappresentare l'autostrada lungo la quale è collocato, un codice ed il nome (univoci all'interno della rete, di solito il nome è quello della località in cui è ubicato), la sua posizione in Km da una delle due località estremi dell'autostrada (sempre la stessa per la stessa autostrada), il volume di traffico massimo sostenibile in entrata ed in uscita (in termini di veicoli/ora), le modalità di pagamento ammesse (in particolare: contanti, carte di credito, bancomat, ViaCard, Telepass), e l'eventuale presenza di un servizio di assistenza ai clienti.

Le tariffe applicate ai veicoli sono espresse in Euro/Km. Queste possono essere diverse per autostrade diverse e per classi di veicoli diverse.

Ogni qualvolta un veicolo entra nella rete autostradale attraverso un casello, il conducente è tenuto a ritirare un tagliando contrassegnato da un codice identificativo. Il sistema associa il tagliando al casello di ingresso, alla classe del veicolo, ed alla data ed ora del passaggio. (Tale tagliando è virtuale in caso il veicolo sia equipaggiato di sistema Telepass, v. seguito.) All'uscita dalla rete, il conducente è tenuto ad inserire il tagliando in suo possesso nell'apposita fessura dell'interfaccia hardware del casello oppure al personale di servizio che provvede a tale inserimento. Il sistema *EasyToll* deve poter calcolare l'importo dovuto applicando la tariffa chilometrica (relativa alla classe del veicolo) per il numero di chilometri percorsi e mostrarla all'automobilista per avviare la procedura di pagamento (per semplicità, in questa sede si ignori tale procedura).

In caso di utilizzo del sistema *Telepass*, le procedure di ingresso ed uscita da un casello avvengono senza l'arresto del veicolo. Ogni dispositivo *Telepass* è univocamente associato ad un cliente abbonato al servizio. All'ingresso, il casello identifica (mediante un segnale radio) il cliente e genera un tagliando virtuale (che viene memorizzato nel sistema) con le stesse informazioni dell'equivalente cartaceo. All'uscita, il casello identifica nuovamente il cliente e calcola l'importo dovuto per il tragitto. Di ogni cliente abbonato al servizio *Telepass* interessa nome, cognome, codice fiscale e coordinate IBAN per l'addebito del conto mensile.)

Il sistema di fatturazione (un sistema esterno a *EasyToll*), dato un cliente *Telepass*, un mese ed un anno, deve poter ottenere il conto di quel cliente nel dato mese del dato anno, contenente, per ogni singolo addebito del mese, il nome dei caselli e la data e l'ora di entrata ed uscita, oltre che l'importo addebitato.)

Per disincentivare gli eccessi di velocità, lungo la rete sono posizionati terminali del sistema *Tutor* equipaggiati da telecamere e di apposito software per il riconoscimento delle targhe dei veicoli. Il sistema *Tutor* permette di rilevare l'eccesso della velocità media calcolata tenendo conto del tempo impiegato da un veicolo per viaggiare lungo un breve tratto delimitato da due terminali *Tutor* (di solito posizionati tra i 10 e 25 Km tra loro). Per ogni coppia di terminali *Tutor* il sistema deve poter memorizzare la sua ubicazione (autostrada, senso di marcia, posizione del terminale di inizio e di quello di fine, in modo analogo alle posizioni dei caselli).

calcolo
importo
pedaggio

calcolo
conto

calcolo
velocità
media

Ogni volta che un veicolo attraversa il terminale primo di una coppia di terminali Tutor (quello di ingresso), EasyToll memorizza la targa e la classe del veicolo, la coppia di terminali che lo sta monitorando, la data e l'ora.) Quando un veicolo attraversa il terminale Tutor di uscita, il sistema calcola la sua velocità media nel breve tratto controllato. Se questa è minore o uguale alla velocità massima consentita (un parametro del sistema che può dipendere anche dalla classe del veicolo oggetto del controllo), il sistema elimina immediatamente i dati del passaggio, altrimenti li mantiene di modo che possano essere acceduti dal personale della Polizia Stradale.

tutore
targa
veicolo



Sapienza Università di Roma
Facoltà di Ing. dell'Informazione, Informatica e Statistica, Laurea in Informatica
Insegnamento di **Basi di Dati, Modulo 2**
Prof. Toni Mancini
Dipartimento di Informatica
<http://tmancini.di.uniroma1.it>

Esame BD2.Esame.Risposte – Modulo risposte prova scritta

Dati dello studente e dell'esame

Cognome e nome: **BIANCO SIMONE** Matricola: **11**
Data: **25/08/23**

Corso di laurea e canale di appartenenza:

- Laurea in Informatica, canale 1 (A-L, Prof. M. Mancini)
- Laurea in Informatica, canale 2 (M-Z, Prof.ssa M. De Marsico)
- Laurea in Informatica in Modalità Teledidattica Unitelma Sapienza

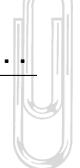
Firma di un membro della Commissione per
avvenuta identificazione:
.....

Rinuncia alla prova

- Desidero rinunciare a questa prova d'esame. Firma:



Questo modulo è ottimizzato per la stampa fronte-retro



Istruzioni e regole d'esame

Prima dell'esame

- Stampare questo modulo, preferibilmente fronte-retro, e rilegarlo con un fermaglio rimovibile, come quello disegnato in alto
- Compilare il frontespizio con i propri dati, come richiesto
- Scrivere la propria matricola nello spazio apposito nella parte alta di tutte le pagine

Durante l'esame

- La prova è dimensionata per essere svolta in circa 3 ore. Tuttavia, data la sua natura fortemente progettuale, la Commissione offre agli studenti la più ampia disponibilità di tempo, al fine ovviare ad eventuali (e limitati) errori di analisi/progettazione rilevati più a valle del ciclo di vita.

Il tempo massimo per la consegna è quindi rilassato a 5 ore (il massimo tempo compatibile con le disponibilità di aule).

- Scrivere le risposte negli spazi predisposti sotto le relative domande. Le ultime pagine sono vuote e possono essere usate come minute oppure, se puntate opportunamente, per contenere risposte in caso gli spazi appositi dovessero risultare insufficienti.
- Non è possibile usare alcun tipo di materiale didattico.
- In caso di necessità di ulteriori fogli (in proprio possesso), chiedere preventivamente alla Commissione una nuova procedura di controllo.
- La Commissione può rispondere solo a brevi domande inerenti al testo dei quesiti.
- Tra la seconda e la quarta ora d'esame, gli studenti possono effettuare **brevi pause** (uno studente alla volta) seguendo la seguente procedura:
 1. Alla lavagna è riportata una coda denominata 'Coda prenotazioni pause'. Sia n (un intero) l'elemento in fondo alla coda (si assuma $n = 0$ in caso di coda vuota).
 2. Recarsi alla lavagna ed aggiungere l'intero $n + 1$ come proprio contrassegno in fondo alla coda, seguito da una stringa a propria scelta (ad es., le proprie iniziali).
 3. Se il proprio contrassegno non è l'elemento affiorante della coda, tornare al lavoro in attesa che lo diventi.
 4. Consegnare tutti i fogli di lavoro e il testo d'esame alla Commissione ed uscire.
 5. Al rientro, cancellare il proprio contrassegno dalla coda di modo da permettere al successivo studente prenotato di uscire, e riprendere i fogli prima consegnati.

Al momento della consegna

- Ordinare tutti i fogli che si vuole far valutare e rilegarli con un fermaglio rimovibile. Non includere fogli che la Commissione non deve valutare (ad es., requisiti, minute), ma includere ovviamente il frontespizio.
- Consegnare i fogli ordinati **nelle mani** di un membro della Commissione. **Non lasciare l'aula senza la conferma, da parte della Commissione, del buon esito delle operazioni di consegna.**

In caso di rinuncia

- È possibile rinunciare alla consegna a partire dalla seconda ora d'esame. In caso di rinuncia, consegnare nelle mani della Commissione solo il frontespizio, dopo aver compilato e firmato la sezione dedicata.

Sommario delle domande

Si richiede di progettare l'applicazione descritta dalla specifica dei requisiti effettuando le fasi di Analisi concettuale dei requisiti e di Progettazione logica della base dati e delle funzionalità, utilizzando la metodologia vista nel corso.

In particolare (vengono indicati i tempi suggeriti per i diversi passi chiave):

Parte 1: Analisi concettuale dei requisiti Effettuare la fase di Analisi concettuale dei requisiti producendo lo schema concettuale per l'applicazione, che includa:

- Analisi dei dati (45 minuti; 75 minuti al massimo):
 - un diagramma ER concettuale (*)
 - il relativo dizionario dei dati
 - le specifiche dei domini concettuali non di tipo base
 - eventuali vincoli esterni, espressi utilizzando il linguaggio della logica del primo ordine (*)
- Analisi delle funzionalità:
 - un diagramma UML degli use-case (5 minuti; 10 minuti al massimo)
 - la segnatura di tutte le operazioni di use-case (10 minuti)
 - la specifica delle operazioni di use-case necessarie a modellare i requisiti contrassegnati dalla barra laterale (come quella qui a sinistra) in termini di precondizioni e postcondizioni, utilizzando il linguaggio della logica del primo ordine (*) (30 minuti; 60 minuti al massimo)

Parte 2: Progettazione della base dati e delle funzionalità Effettuare la progettazione della base dati e delle funzionalità a partire dallo schema concettuale prodotto nella Parte 1, ed in particolare eseguire i seguenti passi:

- Progettazione della base dati relazionale con vincoli:
 - Ristrutturazione del diagramma ER concettuale e dei vincoli esterni (20 minuti; 30 minuti al massimo):
 - * scelta del DBMS da utilizzare
 - * progettazione della corrispondenza tra i domini concettuali ed opportuni domini SQL (domini base o utente, oppure realizzati mediante relazioni aggiuntive) supportati dal DBMS scelto
 - * ristrutturazione del diagramma ER concettuale e dei vincoli esterni.
 - Produzione dello schema relazionale della base dati e dei relativi vincoli (*) (30 minuti; 60 minuti al massimo)
- Progettazione delle funzionalità (30 minuti; 45 minuti al massimo):
 - definizione della specifica realizzativa delle operazioni di use-case necessarie a modellare i requisiti contrassegnati dalla barra laterale, in modo conforme alla loro specifica concettuale prodotta nella fase di Analisi, in termini di algoritmi in pseudo-codice e comandi SQL. (*)

(*) Una risposta soddisfacente a questa domanda è condizione *necessaria* (ma non sufficiente) per superare la prova.

Le pagine seguenti contengono le domande specifiche a cui è richiesto rispondere, ulteriori delucidazioni per ogni singolo punto, e spazi per le risposte.

Le pagine da 33 in poi possono essere utilizzate per scrivere minute che non verranno valutate.



Questa pagina è stata intenzionalmente lasciata vuota

1 Analisi concettuale

Domanda 1 (10 minuti) Raffinare la specifica dei requisiti eliminando inconsistenze, omissioni e ridondanze e producendo un elenco numerato di requisiti il meno ambiguo possibile. (La risposta a questa domanda non sarà valutata, ma si consiglia di svolgere accuratamente questo passo, in quanto può facilitare di molto le attività di progetto.)

Risposta

AUTOSTRADA

- CODICE ALFANUM
- NOME (0,1)
- CITTÀ' ESTREMI
- ESTREMO PRINCIPALE

CASELLA

- CODICE (UNICO)
- NOME (UNICO)
- DISTANZA KM DA ESTREMO PRINCIPALE
- VOL TRAFFICO SOSTENIBILE (VEICOLI/ORA)
- MOD PAGAMENTO AMMESSE
- PRESENZA SERV CLIENTI (0,1)

TARIFFE

- ED/RO/km
- DIPENDE DALL'AUTOSTRADA E DALLA CLASSE DEL VEICOLO

VEICOLO

- TARGA
- CLASSE

MODALITÀ PAGAMENTO

- CONTANTI
- CARTA
- BANCOMAT
- VIACARA
- TELEPASS

TABLIANDO

- CODICE
- INGRESSO
- CLASSE VEICOLO
- ISTANTE ING

CLIENTE TELEPASS

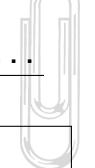
- NAME, COIN
- CF
- COORDINATE IBAN

TUTOR

- AUTOSTRADA
- SENSO MARCIA
- DISTANZA KM DA ESTREMO PRINCIPALE

TELEPASS

- TABLIANDO VIRTUALE
- CLIENTE



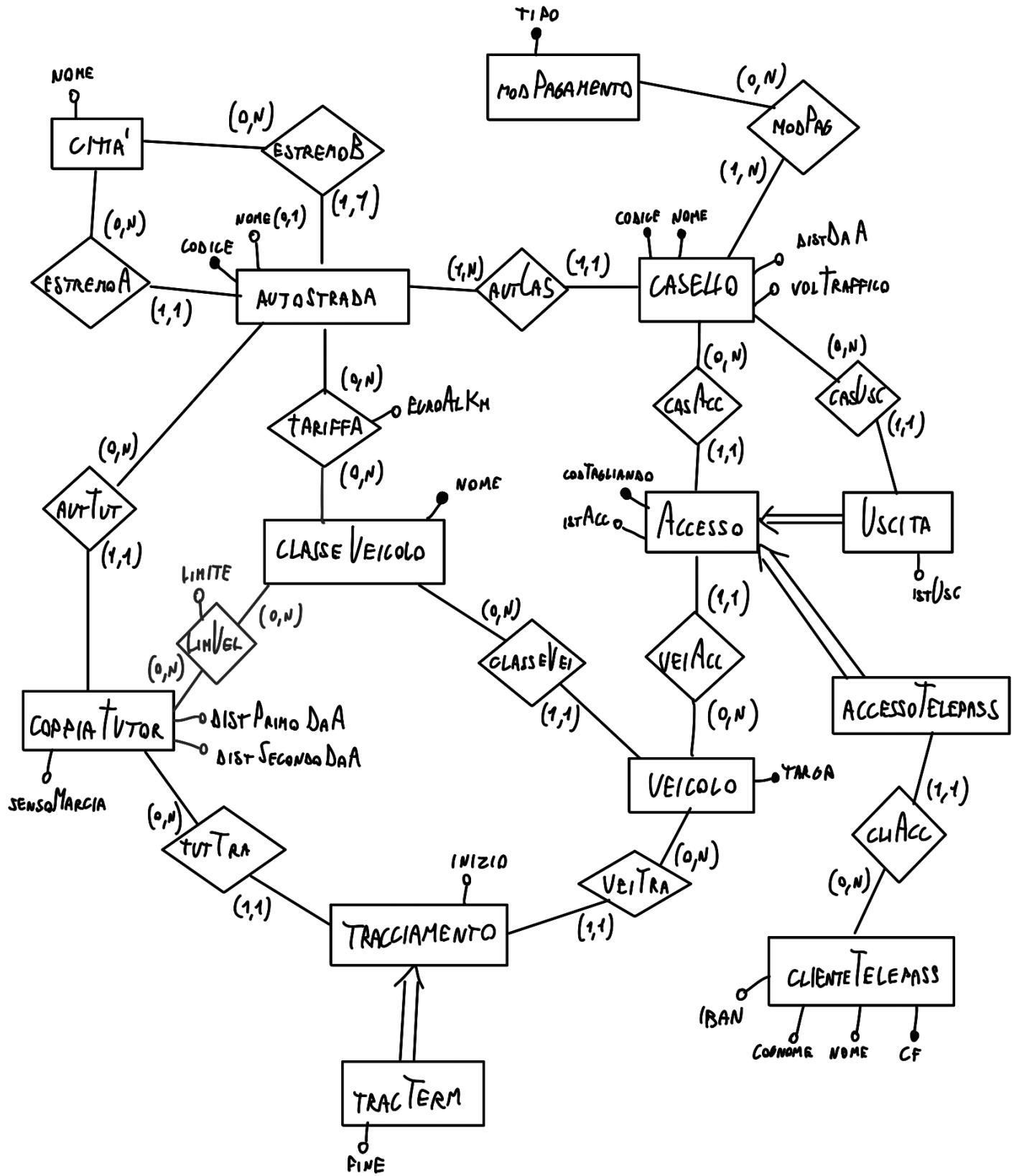
Risposta alla Domanda 1 (segue)

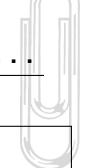
Domanda 2 (45 minuti; 75 minuti al massimo) Proseguire la fase di Analisi Concettuale dei requisiti, producendo un diagramma ER concettuale per l'applicazione, il dizionario dei dati ed eventuali vincoli esterni.

Una risposta soddisfacente a questa domanda è condizione *necessaria* (ma non sufficiente) per superare la prova.

Diagramma ER

Prodotto un diagramma ER concettuale per l'applicazione in termini di entità, relationship, attributi, relazioni is-a, generalizzazioni (disgiunte) complete e non.





Risposta alla Domanda 2 (segue)

Dizionario dei dati Per ogni entità e relationship del diagramma ER **con** attributi o vincoli:

- Definire il dominio e la molteplicità degli attributi (se diversa da (1,1))
- Definire eventuali vincoli esterni in logica del primo ordine estesa con teoria degli insiemi e semantica di mondo reale, usando il seguente alfabeto:
 - Un simbolo di predicato $E/1$ per ogni entità E .
Semantica di $E(x)$: x è una istanza di E .
 - Un simbolo di predicato $D/1$ per ogni dominio D .
Semantica di $D(x)$: x è un valore di D .
 - Un simbolo di predicato r/n ($n > 0$) per ogni relationship n -aria r .
Semantica di $r(x_1, \dots, x_n)$: x_1, \dots, x_n è una istanza di r .
 - Un simbolo di predicato $a/2$ per ogni attributo a di entità
Semantica di $a(x, v)$: uno dei valori dell'attributo a dell'istanza x è v .
 - Un simbolo di predicato $a/(n+1)$ per ogni attributo a di relationship n -aria.
Semantica di $a(x_1, \dots, x_n, v)$: uno dei valori dell'attr. a dell'istanza (x_1, \dots, x_n) della relat. è v .
 - Opportuni simboli di predicato (soggetti a *semantica di mondo reale*) per gestire confronti tra valori di domini numerici o comunque ordinati (tra cui $</2$, $\leq/2$, $>/2$, $\geq/2$).
 - Il predicato di uguaglianza $=/2$ (la cui interpretazione è la relazione che lega ogni elemento del dominio di interpretazione solo con se stesso).
 - Opportuni simboli di costante (soggetti a *semantica di mondo reale*), tra cui *adesso*, interpretato come il valore del dominio DataOra che rappresenta l'istante corrente.

Risposta

<p>[1] Tipo: Entità Relationship (cerchiare)</p> <p>Nome: ... CITTÀ ...</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> <tbody> <tr> <td>NAME</td> <td>STRINGA</td> <td></td> </tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>X • V. AUTOSTRADA . ESTREMI DIVERSI $\forall z, c \text{ AUTOSTRADA}(z) \wedge \text{CITTÀ}'(c) \wedge \text{ESTREMOA}(z, c)$ $\rightarrow \exists \text{ESTREMOB}(z, c)$</p> <p>R • V. AUTOSTRADA . CASELLI $\forall u, c, c', a \text{ USCITA}(u) \wedge \text{CASELLA}(u, c) \wedge \text{CASUsc}(u, c')$ $\wedge \text{autCas}(c, a) \rightarrow \text{autCas}(c', a)$</p>	attributo	dominio	moltep. (*)	NAME	STRINGA		<p>[2] Tipo: Entità Relationship (cerchiare)</p> <p>Nome: ... AUTOSTRADA ...</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> <tbody> <tr> <td>CODICE</td> <td>STRINGA</td> <td></td> </tr> <tr> <td>NAME</td> <td>STRINGA</td> <td>(a,1)</td> </tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>X • V. AUTOSTRADA . ESTREMI DIVERSI $\forall z, c \text{ AUTOSTRADA}(z) \wedge \text{CITTÀ}'(c) \wedge \text{ESTREMOA}(z, c)$ $\rightarrow \exists \text{ESTREMOB}(z, c)$</p> <p>R • V. AUTOSTRADA . CASELLI $\forall u, c, c', a \text{ USCITA}(u) \wedge \text{CASELLA}(u, c) \wedge \text{CASUsc}(u, c')$ $\wedge \text{autCas}(c, a) \rightarrow \text{autCas}(c', a)$</p>	attributo	dominio	moltep. (*)	CODICE	STRINGA		NAME	STRINGA	(a,1)
attributo	dominio	moltep. (*)														
NAME	STRINGA															
attributo	dominio	moltep. (*)														
CODICE	STRINGA															
NAME	STRINGA	(a,1)														

<input type="checkbox"/> 3	Tipo: Entità	Relationship (cerchiare)
Nome: ...	CASELLO	
attributo	dominio	moltep. (*)
CODICE	STRINGA	
NAME	STRINGA	
DISTANZA	REAL>0	
VOLTRAFFICO	REAL>0	

(*) solo se diversa da (1,1)

Vincoli:

- V. CASELLO. VOLTRAFFICO
 $\forall c, t, h \text{ CASELLO}(c) \wedge \text{VOLTRAFFICO}(c, t) \wedge \text{ORA}(h) \rightarrow$
 $\text{NUNVEICOLI}(c, o) \leq t$

<input type="checkbox"/> 5	Tipo: Entità	Relationship (cerchiare)
Nome: ...	MODPAGAMENTO	
attributo	dominio	moltep. (*)
TIPO	{'co', 'cc', 'b', 'v', 't'}	

(*) solo se diversa da (1,1)

Vincoli:

<input type="checkbox"/> 4	Tipo: Entità	Relationship (cerchiare)
Nome: ...	ACCESSO	
attributo	dominio	moltep. (*)
COSTABILIANO	STRINGA	
1STAcc	DATAORA	

(*) solo se diversa da (1,1)

Vincoli:

- V. ACCESSO. DISGIUNTI
 $\nexists v, a, a', i, i' \text{ VEICOLO}(v) \wedge \text{VEIAcc}(a, v) \wedge$
 $\text{VEIAcc}(a', v) \wedge a \neq a' \wedge \text{1STAcc}(a, i) \wedge$
 $\text{1STAcc}(a', i') \rightarrow \nexists t \text{ DATAORA}(t) \wedge$
 $(i \leq t \wedge (\nexists f \text{ 1STUsc}(a, f) \rightarrow t \leq f)) \wedge$
 $(i' \leq t \wedge (\nexists f' \text{ 1STUsc}(a', f') \rightarrow t \leq f'))$

<input type="checkbox"/> 6	Tipo: Entità	Relationship (cerchiare)
Nome: ...	USCITA	
attributo	dominio	moltep. (*)
1STUsc	DATAORA	

(*) solo se diversa da (1,1)

Vincoli:

- V. USCITA. CONTINUITA'
 $\nexists a, i \text{ ACCESSO}(a) \wedge \text{1STAcc}(a, i) \rightarrow$
 $(\nexists f \text{ 1STUsc}(a, f) \rightarrow i < f)$
- V. USCITA. DIVERSITÀAccesso
 $\nexists a, c \text{ ACCESSO}(a) \wedge \text{casAcc}(a, c) \rightarrow$
 $(\text{USCITA}(a) \rightarrow \nexists \text{ casUsc}(a, c))$

7	Tipo: <input checked="" type="checkbox"/> Entità Relationship (cerchiare)	Nome: <u>ACCESSO TELEPASS</u>
	attributo	dominio moltep. (*)
<hr/>		
(*) solo se diversa da (1,1)		

Vincoli:

R • V.ACCESSO TELEPASS.LEGALE

$$\forall c, a \text{ ACCESSO TELEPASS}(a) \wedge \text{CASAcc}(a, c) \\ \rightarrow \exists p \text{ METPas}(c, p) \wedge \text{TIPO}(p, 'T')$$

9	Tipo: <input checked="" type="checkbox"/> Entità Relationship (cerchiare)	Nome: <u>CLIENTE TELEPASS</u>
	attributo	dominio moltep. (*)
<hr/>		
(*) solo se diversa da (1,1)		

Vincoli:

8	Tipo: <input checked="" type="checkbox"/> Entità Relationship (cerchiare)	Nome: <u>VEICOLO</u>
	attributo	dominio moltep. (*)
<hr/>		
(*) solo se diversa da (1,1)		

Vincoli:

10	Tipo: <input checked="" type="checkbox"/> Entità Relationship (cerchiare)	Nome: <u>CLASSE VEICOLO</u>
	attributo	dominio moltep. (*)
<hr/>		
(*) solo se diversa da (1,1)		

Vincoli:

11	Tipo: Entità Relationship (cerchiare)
Nome: ...	TARIFFA
attributo	dominio
EURO/km	DENARO
(*) solo se diversa da (1,1)	
Vincoli:	
T	<ul style="list-style-type: none"> • V. TARIFFA. ESISTENZA $\forall a, c, a', v, c' \text{ AUTOSTRADA}(a) \wedge \text{VEI}(as(a, a)) \wedge$ $\text{casAcc}(a', c) \wedge \text{VEIAcc}(a', v) \wedge \text{classeVe}(v, c')$ $\rightarrow \text{TARIFFA}(a, c')$

13	Tipo: Entità Relationship (cerchiare)
Nome: ...	TRACCIMENTO
attributo	dominio
INIZIO	DATATURA
(*) solo se diversa da (1,1)	
Vincoli:	
	<ul style="list-style-type: none"> • V. TRACCIMENTO. DISGIUNTI $\forall v, t, t', i, i' \text{ VEICOLO}(v) \wedge \text{VEITRA}(t, v) \wedge$ $\text{VEITRA}(v, t') \wedge t \neq t' \wedge \text{INIZIO}(t, i) \wedge \text{INIZIO}(t', i')$ $\rightarrow \nexists t \text{ DATATURA}(t) \wedge (i \leq t \wedge (\forall x \text{ FINE}(t, x) \rightarrow t < x))$ $\wedge (i \leq t \wedge (\forall x' \text{ FINE}(t', x') \rightarrow t' < x'))$

12	Tipo: Entità Relationship (cerchiare)
Nome: ...	TRAC TERM
attributo	dominio
FINE	DATATURA
(*) solo se diversa da (1,1)	
Vincoli:	
X	<ul style="list-style-type: none"> • V. TRAC TERM. CONTINUITÀ $\forall t, i \text{ TRACCIMENTO}(t) \wedge \text{INIZIO}(t, i)$ $\rightarrow (\forall x \text{ FINE}(t, x) \rightarrow i < x)$

14	Tipo: Entità Relationship (cerchiare)
Nome: ...	COPPIA TUTOR
attributo	dominio
SESSOMARCA	$\{A \rightarrow B, B \rightarrow A\}$
DISTANZAPrimoMA	REALE70
DISTANZASecondoMA	REALE70
(*) solo se diversa da (1,1)	
Vincoli:	

15 Tipo: Entità | **Relationship** (cerchiare)

Nome: LIM~~VEL~~

attributo	dominio	moltep. (*)
<u>LIMITE</u>	<u>REALE>0</u>	

(*) solo se diversa da (1,1)

Vincoli:

- V.LIM~~VEL~~.TRACCIAMENTO

$$\forall t_r, t, v, c \quad TRACCIAMENTO(t_r) \wedge TUTTRA(t_r, t) \\ \wedge VERTRA(t_r, v) \wedge CLASSEVEL(v, c) \rightarrow LIM~~VEL~~(t, c)$$

17 Tipo: Entità | Relationship (cerchiare)

Nome:

attributo	dominio	moltep. (*)

(*) solo se diversa da (1,1)

Vincoli:

16 Tipo: Entità | Relationship (cerchiare)

Nome:

attributo	dominio	moltep. (*)

(*) solo se diversa da (1,1)

Vincoli:

18 Tipo: Entità | Relationship (cerchiare)

Nome:

attributo	dominio	moltep. (*)

(*) solo se diversa da (1,1)

Vincoli:

Ulteriori vincoli esterni, specifica di eventuali operazioni ausiliarie invocate da tali vincoli, e specifica dei domini concettuali non di tipo base

- DOMINIO CONFIS: STRINGA 16 CARATTERI SECONDO STANDARD
- DOMINIO IBAN: STRINGA 23 SECONDO STANDARD
- DOMINIO TAAGA: STRINGA 10 CARATTERI SECONDO STANDARD
- DOMINIO DENARO:
 - VALUTA: STRINGA 3 CARATTERI SECONDO STANDARD
 - IMPORTO: REALE > 0
- NUMVEICOLI (c: CASELLA, h: INTERO $[0, 23]$): INTERO > 0

PRE:

POST:

$$A = \left\{ v \mid \begin{array}{l} VEICOLO(v) \wedge (\exists a, i, o_i \text{ ACCESSO}(a) \wedge \text{USCITA}(a) \\ \wedge \text{ISTAcc}(a, i) \wedge \text{ORA}(i, o_i) \wedge \text{HT}(o_i, h)) \end{array} \right\}$$

$$V = \left\{ v \mid \begin{array}{l} VEICOLO(v) \wedge (\exists u, f, o_f \text{ USCITA}(u) \wedge \\ \wedge \text{ISTRUsc}(u, f) \wedge \text{ORA}(f, o_f) \wedge \text{HT}(o_f, h)) \end{array} \right\}$$

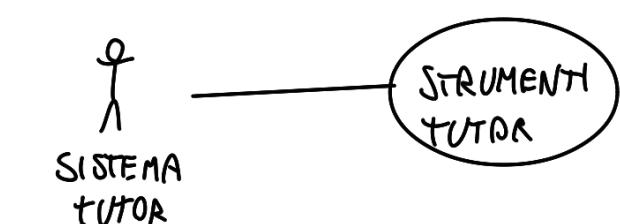
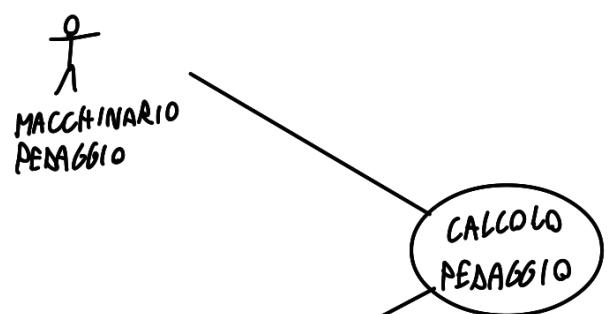
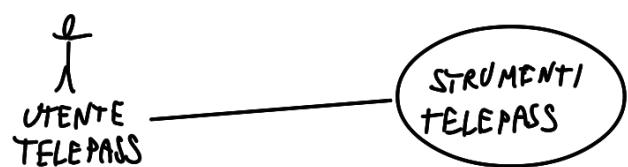
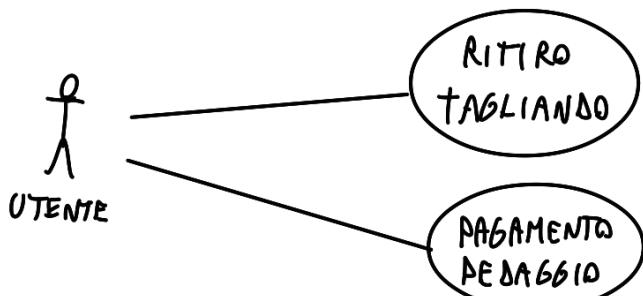
$$\text{RESULT} = |A| + |V|$$

• V. TRACCIAIMENTO DURANTE ACCESSO

$$\forall t, i, v \text{ TRACCIAIMENTO}(t) \wedge \text{INIZIO}(t, i) \wedge \text{VISITRAC}(t, v) \rightarrow (\exists a, i' \text{ ACCESSO}(a) \wedge \text{ISTAcc}(a, i) \\ \wedge (i' \leq i) \wedge (\forall f \text{ FINE}(t, f) \rightarrow (\exists f' \text{ ISTRUsc}(a, f') \rightarrow f \leq f')))$$

Risposta alla Domanda 2 (segue)

Domanda 3 (5 minuti; 10 minuti al massimo) Proseguire la fase di Analisi Concettuale dei requisiti, producendo un diagramma UML degli use-case che definisca ad alto livello tutte le funzionalità richieste al sistema.

Risposta

Domanda 4 (10 minuti) Proseguire la fase di Analisi Concettuale dei requisiti definendo le operazioni degli use-case.

In particolare, per ogni use-case definito nella risposta alla **Domanda 3** definire la **segnatura** di tutte le operazioni che lo compongono, in termini di nome dell'operazione, nomi e dominio concettuale degli argomenti, dominio concettuale dell'eventuale valore di ritorno.

1 Specifica use-case: RITRA TAGLIANDO (nome use-case)

Operazioni dello use-case:

RITRATAGLIANDO (v: VEICOLO, c: CASELLO): ACCESSO

2 Specifica use-case: PAGAMENTO PEDAGGIO (nome use-case)

Operazioni dello use-case:

PAGA PEDAGGIO (a: ACCESSO)

3 Specifica use-case: STRUMENTI TELEPASS (nome use-case)

Operazioni dello use-case:

CALCOLACONTOTELEPASS (u: CLIENTETELEPASS, m: INTERO [1, 12], a: INTERO 7, 2000):
 $(c: CASELLO, i: DATAORA, f: DATAORA, d: DENARO) (0, N)$

ACCEDICONTELEPASS (u: CLIENTETELEPASS, c: CASELLO): ACCESSOTELLEPASS

ESCIICONTELEPASS (u: CLIENTETELEPASS, a: ACCESSOTELLEPASS)

4 Specifica use-case: CALCOLA COSTO PEDAGGIO (nome use-case)

Operazioni dello use-case:

■ CALCOLACOSTOPEDAGGIO(a:ACCESSO, cU:CASELLA):DENARO

5 Specifica use-case: STRUMENTI TUTOR (nome use-case)

Operazioni dello use-case:

■ AVIA TRACCIAIMENTO(t:LOMIA TUTOR, v:VEICOLO)

■ TERMINA TRACCIAIMENTO(t:LOMIA TUTOR, v:VEICOLO)

6 Specifica use-case: (nome use-case)

Operazioni dello use-case:

7 Specifica use-case: (nome use-case)

Operazioni dello use-case:

Domanda 5 (30 minuti; 60 minuti al massimo) Proseguire la fase di Analisi Concettuale dei requisiti producendo le specifiche concettuali per le operazioni di use-case, **limitandosi** a quelle necessarie a modellare i requisiti contrassegnati dalla barra laterale (come quella qui a sinistra). In particolare, per ogni operazione, definire segnatura, precondizioni e postcondizioni utilizzando il linguaggio della logica del primo ordine. Si assuma lo stesso vocabolario definito alla [Domanda 2](#).

Una risposta soddisfacente a questa domanda è condizione *necessaria* (ma non sufficiente) per superare la prova.

Risposta

CALCOLA CANTO TELEPASS ($u: CUENTTELEPASS$, $m: \text{INTERO}[1, 12]$, $a: \text{INTERO}[-2000, 2000]$):
 $(n_a: \text{STRINGA}, n_u: \text{STRINGA}, i: \text{DATAORA}, f: \text{DATAORA}, d: \text{DENARO}) (0, N)$

PRE:

POST:

$$C = \left\{ (n_a, n_u, i, f, d) \mid \begin{array}{l} \exists c_a, cu, a, da, du, v, c, a', e, di, ai, ni \\ \text{casello}(c_a) \wedge \text{CASELLO}(cu) \wedge \text{NAME}(c_a, n_a) \wedge \text{NAME}(cu, n_u) \wedge \\ \text{ACCESSOTELEPASS}(a) \wedge \text{USCITA}(a) \wedge \text{CLI Acc}(a, u) \wedge \text{ISTAcc}(a, i) \\ \wedge \text{DATA}(i, di) \wedge \text{ANNO}(di, ai) \wedge \text{MESE}(di, ni) \wedge \text{ISTUSC}(a, f) \\ \wedge \text{CASAcc}(a, ca) \wedge \text{CASUSC}(a, cu) \wedge \text{DISTDA}(ca, da) \wedge \text{DISTDA}(cu, du) \\ \wedge \text{VEI Acc}(a, v) \wedge \text{CLASSE VEI}(v, c) \wedge \text{TARIFFA}(a', c) \wedge \\ \text{EUROALKM}(a', c, e) \wedge (d = |da - du| \cdot e) \wedge (ni = m \wedge ai = a) \end{array} \right\}$$

RESULT = C

CALCOLA COSTO PEDAGGIO ($a: \text{ACCESSO}$, $cu: \text{CASELLO}$): DENARO

PRE: $\neg \text{USCITA}(a) \wedge \neg \text{CASAcc}(a, cu) \wedge$
 $(\exists a' \text{ AUTOSTRADA}(a') \wedge \text{AUTLAS}(ca, a') \wedge \text{AUTLAS}(cu, a'))$

POST:

Sono da, du, e tali che

$\exists ca, a', c, v$
 $\text{CASELLO}(ca) \wedge \text{CASAcc}(a, ca) \wedge \text{AUTOSTRADA}(a') \wedge$
 $\text{AUTLAS}(ca, a') \wedge \text{DISTDA}(ca, da) \wedge \text{DISTDA}(ca, du) \wedge$
 $\text{VEI Acc}(a, v) \wedge \text{CLASSE VEI}(v, c) \wedge \text{TARIFFA}(a', c) \wedge$
 $\text{EUROALKM}(a', c, e)$

RESULT = $|da - du| \cdot e$

Risposta alla Domanda 5 (segue)

$\text{AVVIA TRACCIAMENTO } (t: \text{COPPIA TUTOR}, v: \text{VEICOLO})$

PRE: $\exists t_x \text{ TRACCIAMENTO}(t_x) \wedge \exists \tau \text{ TRACTERM}(t_x) \wedge \text{VEITRA}(t_x, v) \wedge \text{TUTORTRA}(t_x, t)$
 $\wedge (\exists a, i, c, a' d \text{ ACCESSO}(d) \wedge \text{USCITA}(d) \wedge \text{ISTRACC}(a, i) \wedge i < \text{ADESSO}$
 $\wedge \text{CASACC}(a, c) \wedge \text{AUTLAS}(c, a) \wedge \text{AUTTUT}(t, a) \wedge \text{VEIACC}(a, v) \wedge$
 $\text{CLASSEVEI}(v, d) \wedge \text{LIMVEL}(t, d))$

POST:

- NUOVO ELEMENTO: d
- NUOVI PREDICATI:
 $\text{TRACCIAMENTO}(d), \text{VEITRA}(d, v), \text{INIZIA}(d, \text{ADESSO}), \text{TUTORTRA}(t_x, t)$

$\text{TERMINA TRACCIAMENTO } (t: \text{COPPIA TUTOR}, v: \text{VEICOLO})$

PRE: $\exists t_x \text{ TRACCIAMENTO}(t_x) \wedge \exists \tau \text{ TRACTERM}(t_x) \wedge \text{VEITRA}(t_x, v) \wedge \text{TUTORTRA}(t_x, t)$
 $\wedge (\exists a, i, c, a' d \text{ ACCESSO}(d) \wedge \text{USCITA}(d) \wedge \text{ISTRACC}(a, i) \wedge i < \text{ADESSO}$
 $\wedge \text{CASACC}(a, c) \wedge \text{AUTLAS}(c, a) \wedge \text{AUTTUT}(t, a) \wedge \text{VEIACC}(a, v) \wedge$
 $\text{CLASSEVEI}(v, d) \wedge \text{LIMVEL}(t, d))$

POST:

Sicurezza d, d', i, l

$\exists t_x, c \text{ TRACCIAMENTO}(t_x) \wedge \exists \tau \text{ TRACTERM}(t_x) \wedge \text{INIZIO}(t_x, i) \wedge$
 $\text{VEITRA}(t_x, v) \wedge \text{TUTORTRA}(t_x, t) \wedge \text{CLASSEVEI}(v, c) \wedge \text{LIMVEL}(t, c)$
 $\wedge \text{LIMITE}(t, c, l) \wedge \text{DISTPRIMOAA}(t, d) \wedge \text{DISTSECONDOAA}(t, d')$

$$M = \frac{|d - d'|}{(\text{ADESSO} - i) \cdot 360}$$

- MODIFICHE AI PREDICATI:

$(M > l \rightarrow \text{TRACTERM}(t) \wedge \text{FINE}(t, \text{ADESSO}))$
 \wedge
 $(M < l \rightarrow \neg \text{TRACCIAMENTO}(t))$

2 Progettazione della base dati e delle funzionalità

Domanda 6 (20 minuti; 30 minuti al massimo) Iniziare la fase di progettazione logica della base di dati decidendo il DBMS da utilizzare e ristrutturando lo schema ER concettuale, il dizionario dei dati e i vincoli esterni. In particolare:

- progettare una corrispondenza tra i domini concettuali ed opportuni domini SQL (domini base o utente, oppure realizzati mediante relazioni aggiuntive) supportati dal DBMS scelto
- eliminare attributi multivale o composti
- eliminare relazioni is-a e generalizzazioni
- definire un identificatore primario per ogni entità
- valutare se e come aggiungere ridondanza in maniera controllata
- ristrutturare i vincoli esterni per renderli consistenti con la struttura del nuovo diagramma.

Descrivere brevemente le principali scelte effettuate.

DBMS da utilizzare **POSTGRESQL**

Corrispondenza tra domini concettuali e domini supportati dal DBMS

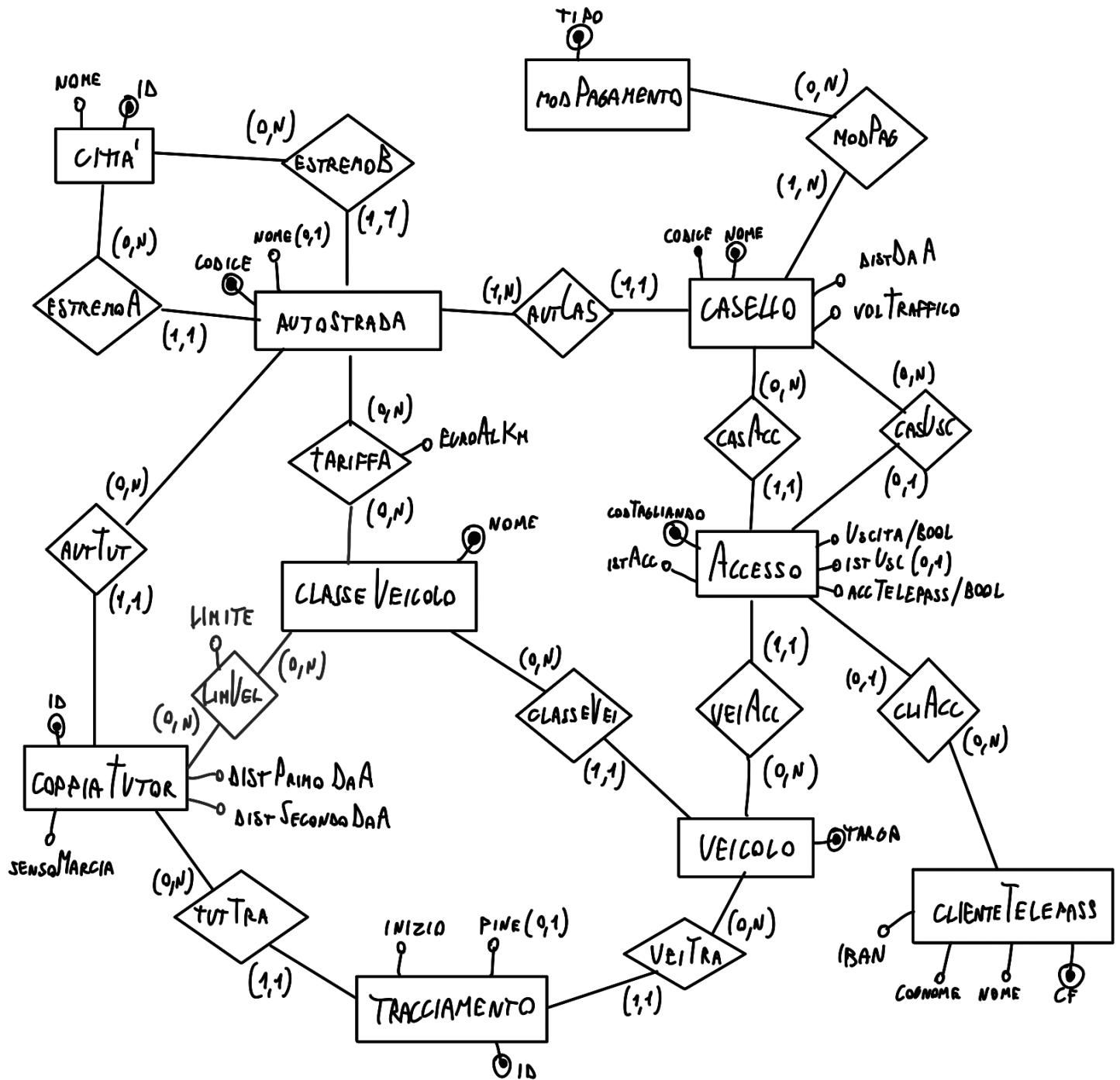
- CREATE DOMAIN INT-GZ AS INTEGER CHECK (VALUE > 0)
- CREATE DOMAIN INT-GEZ AS INTEGER CHECK (VALUE >= 0)
- CREATE DOMAIN REAL-GZ AS REAL CHECK (VALUE > 0)
- CREATE DOMAIN REAL-GEZ AS REAL CHECK (VALUE >= 0)
- CREATE DOMAIN STR AS VARCHAR (200)
- CREATE DOMAIN CODFIS AS CHAR (16) CHECK (ISVALID (F (VALUE)))
- CREATE DOMAIN IBAN AS CHAR (23) CHECK (ISVALIDIBAN (VALUE))
- CREATE DOMAIN TARGA AS CHAR (10) CHECK (ISVALIDTARGA (VALUE))
- CREATE TYPE DENARO AS (

VALUTA CHAR (3),

IMPORTO REAL - GEZ

)
- CREATE TYPE TIPO_PAGAM AS ENUM ('CA', 'CC', 'B', 'V', 'T')
- CREATE TYPE SENSO_MARCIA AS ENUM ('A->B', 'B->A')

Diagramma ER ristrutturato



Breve descrizione delle scelte effettuate durante la ristrutturazione

Vincoli esterni introdotti o modificati durante la fase di ristrutturazione
 (si omettano i vincoli esterni la cui formulazione è rimasta identica a seguito della ristrutturazione)

• **V. ACCESSO. USCITA**

$$\forall a \text{ ACCESSO}(a) \rightarrow (\exists c, \nexists \text{ CASUsc}(a, c) \wedge \text{ISTUsc}(a, \perp) \Leftrightarrow \text{USCITA}(a, \text{TRUE}))$$

• **V. ACCESSO. ACCTELEPASS**

$$\forall a \text{ ACCESSO}(a) \rightarrow (\exists c \text{ CASAcc}(a, c) \Leftrightarrow \text{ACCTELEPASS}(a, \text{TRUE}))$$

• **V. USCITA. DIVERSADAAccesso**

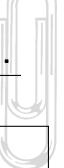
$$\forall a, c \text{ ACCESSO}(a) \wedge \text{CASAcc}(a, c) \rightarrow (\text{USCITA}(a, \text{TRUE}) \rightarrow \exists c' \text{ CASUsc}(a, c'))$$

• **V. ACCESSOTELEPASS. LEGALE**

$$\forall c, a \text{ ACCESSO}(a) \wedge \text{ACCTELEPASS}(a, \text{TRUE}) \wedge \text{CASAcc}(a, c) \rightarrow \exists p \text{ METPas}(c, p) \wedge \text{TIPo}(p, 'T')$$

• **V. AUTOSTRADA. CASELLI**

$$\forall u, c, c', a \text{ ACCESSO}(u) \wedge \text{USCITA}(u, \text{TRUE}) \wedge \text{CASAcc}(u, c) \wedge \text{CASUsc}(u, c') \wedge \text{AutCas}(c, a) \rightarrow \text{AutCas}(c', a)$$



Risposta alla Domanda 6 (segue)

Domanda 7 (30 minuti; 60 minuti al massimo) Proseguire la fase di progettazione logica della base di dati producendo lo schema relazionale della base dati e i relativi vincoli a partire dallo schema ER ristrutturato.

Una risposta soddisfacente a questa domanda è condizione *necessaria* (ma non sufficiente) per superare la prova.

1 Relazione	<u>CITTA'</u>	(nome)	Derivante da:	entità	relationship (cerchiare)
Attributi	<u>ID</u>	<u>NOME</u>			
Domini	INTEGER	STR			

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

SERIAL : ID

La relazione accorda le relazioni che implementano le seguenti relationship:

2 Relazione	<u>AUTOSTRADA</u>	(nome)	Derivante da:	entità	relationship (cerchiare)
Attributi	<u>CODICE</u>	<u>NOME</u> *	<u>ESTREMO A</u>	<u>ESTREMO B</u>	
Domini	STR	STR	INTEGER	INTEGER	

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

FK: (ESTREMO A) REF CITTA'(ID)

ENN: ESTREMO A ≠ ESTREMO B

FK: (ESTREMO B) REF CITTA'(ID)

INC: (CODICE) ⊆ CASELLO(AUTOSTRADA)

La relazione accorda le relazioni che implementano le seguenti relationship: ... ESTREMO A, ESTREMO B

3 Relazione	<u>CASELLO</u>	(nome)	Derivante da:	entità	relationship (cerchiare)
Attributi	<u>NOME</u>	<u>CODICE</u>	<u>DISTDA A</u>	<u>VOLTRAFFICO</u>	<u>AUTOSTRADA</u>
Domini	STR	STR	REAL_62	REAL_62	STR

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

CHIAVE: CODICE

INC: (NOME) ⊆ MODPAG(CASELLO)

FK: (AUTOSTRADA) REF AUTOSTRADA(CODICE)

La relazione accorda le relazioni che implementano le seguenti relationship: ... NUTCAS

4 Relazione	<u>MODPAGAMENTO</u>	(nome)	Derivante da:	entità	relationship (cerchiare)
Attributi	<u>TIPO</u>				
Domini	TIPO_PAGAM				

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

5 Relazione	<u>MODPAG</u>	(nome)	Derivante da:	entità	relationship (cerchiare)
Attributi	<u>CASELLO</u>	<u>MOD</u>			
Domini	STR	TIPO_PAGAM			

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

FK: (CASELLO) REF CASELLO(NOME)

FK: (MOD) REF MODPAGAMENTO(TIPO)

La relazione accorda le relazioni che implementano le seguenti relationship:

6 Relazione ...ACCESO.....	(nome)	Derivante da: entità relationship (cerchiare)
Attributi	<u>COSTOLIANDO</u> CASAcc	<u>CASUSC</u> *
Domini	STR	STR
Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *		
Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):		
FK: (CASAcc) REF CASELLO (none) ENN: USCITA = TRUE \Leftrightarrow ISTUSC ≠ NULL \wedge CASUSC ≠ NULL		
FK: (CASUSC) REF CASELLO (none) ENN: ACCTELEPASS = TRUE \Leftrightarrow CLIENTE ≠ NULL		
FK: (VEICOLO) REF VEICOL0 (TARGA) ENN: ISTUSC ≠ NULL \rightarrow ISTAcc < ISTUSC ENN: CASUSC ≠ NULL \rightarrow CASUSC ≠ CASAcc		
La relazione accorda le relazioni che implementano le seguenti relationship: casAcc, casUSC, veiAcc, cliaAcc		

7 Relazione ...VEICOLO.....	(nome)	Derivante da: entità relationship (cerchiare)
Attributi	<u>TARGA</u>	<u>CLASSE</u>
Domini	STR	STR
Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *		
Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):		
FK: (CLASSE) REF CLASSEVEICOLO (none)		
La relazione accorda le relazioni che implementano le seguenti relationship: classeVe		

8 Relazione ...CLIENTETELEPASS..	(nome)	Derivante da: entità relationship (cerchiare)
Attributi	<u>CF</u>	<u>NOME</u>
Domini	COFIS	STR
Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *		
Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):		
La relazione accorda le relazioni che implementano le seguenti relationship:		

9 Relazione ...CLASSEVEICOLO..	(nome)	Derivante da: entità relationship (cerchiare)
Attributi	<u>NOME</u>	
Domini	STR	
Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *		
Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):		
La relazione accorda le relazioni che implementano le seguenti relationship:		

10 Relazione ...TARIFFE.....	(nome)	Derivante da: entità relationship (cerchiare)
Attributi	<u>AUTOSTRADA</u>	<u>CLASSE</u>
Domini	STR	STR
Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *		
Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):		
FK: (AUTOSTRADA) REF AUTO STRADA (codice)		
FK: (CLASSE) REF CLASSEVEICOLO (none)		
La relazione accorda le relazioni che implementano le seguenti relationship:		

11 Relazione .TRNCIAMENTO... (nome) Derivante da: entità | relationship (cerchiare)

Attributi	<u>ID</u>	VEICOLO	TUTORS	<u>INIZIO</u>	FINE *		
Domini	INTEGER	TARGA	INTEGER	TIMESTAMP	TIMESTAMP		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

SERIAL: ID
FK: (VEICOLO) REF VEICOLO (TARGA)
FK: (TUTORS) REF COPPIATUTOR (ID)

ENV: FINE ≠ NULL → INIZIO < FINE

La relazione accorda le relazioni che implementano le seguenti relationship: ... tutta, veitra

12 Relazione .COPPIATUTOR... (nome) Derivante da: entità | relationship (cerchiare)

Attributi	<u>ID</u>	AUTOSTRADA	DISTANZIA	DISTSECONDA	SENSOMARCA		
Domini	INTEGER	STR	REAL_62	REAL_62	SENSOMARCA		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

SERIAL: ID
FK: (AUTOSTRADA) REF AUTOSTRADA (CODICE)

La relazione accorda le relazioni che implementano le seguenti relationship: ... autut

13 Relazione ...LIMVEL..... (nome) Derivante da: entità | relationship (cerchiare)

Attributi	<u>TUTORS</u>	<u>CLASSE</u>	LIMITE				
Domini	INTEGER	STR	REAL_62				

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

FK: (TUTORS) REF COPPIATUTOR (ID)
FK: (CLASSE) REF CLASSEVEICOLO (NAME)

La relazione accorda le relazioni che implementano le seguenti relationship:

14 Relazione (nome) Derivante da: entità | relationship (cerchiare)

Attributi							
Domini							

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

15 Relazione (nome) Derivante da: entità | relationship (cerchiare)

Attributi							
Domini							

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

16	Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi		
Domini		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

17	Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi		
Domini		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

18	Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi		
Domini		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

19	Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi		
Domini		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

20	Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi		
Domini		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

Ulteriori vincoli esterni

Per ogni ulteriore vincolo esterno (non ancora espresso perché non definibile mediante vincoli di chiave, foreign key, ennupla, dominio, inclusione), progettare un trigger che lo implementi, definendo: (a) gli eventi da intercettare (inserimento, modifica, eliminazione di ennuple); (b) quando intercettare tali eventi (appena prima o subito dopo l'evento intercettato); (c) la relativa funzione in pseudo-codice con SQL immerso che implementa il controllo del vincolo.

• T. ACCESSO TELEPASS. LEGALE

- INSEGNAMENTO IN Accesso

- PRE-OPERAZIONE

```
isValid = (NEW.ACCTELEPASS = TRUE AND
           EXISTS (SELECT *
                    FROM CASELLO C, METPAG MP
                   WHERE NEW.CASACC = C.NOME AND
                         MP.CASELLO = C.NOME AND
                         MP.METODO = 'T'))
```

```
IF isValid
    COMMIT
ELSE
    GENERATE ERRORE
    REVERT
```

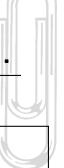
• V. TARIFFA. ESISTENZA

- INSEGNAMENTO IN ACCESSO

- PRE-OPERAZIONE

```
isValid = (EXISTS (SELECT *
                     FROM CASELLO C, VEICOLO V, TARIFFA T
                    WHERE NEW.CASACC = C.NOME AND NEW.VEICOLO = V.TARGA
                          AND V.CLASSE = T.CLASSE AND T.AUTOSTRADA = C.AUTOSTRADA))
```

```
IF isValid
    COMMIT
ELSE
    GENERATE ERRORE
    REVERT
```



Risposta alla Domanda 7 (segue)

Domanda 8 (30 minuti; 45 minuti al massimo) Proseguire la fase di progettazione dell'applicazione producendo le specifiche realizzative delle operazioni di use-case definite per modellare i requisiti contrassegnati dalla barra laterale della specifica dei requisiti.

In particolare, per ogni operazione definire la segnatura, in termini di nome dell'operazione, nomi e dominio SQL degli argomenti, dominio SQL dell'eventuale valore di ritorno, e un algoritmo in pseudo-codice con SQL immerso che verifichi le precondizioni e garantisca il raggiungimento delle postcondizioni definite in fase di Analisi.

Una risposta soddisfacente a questa domanda è condizione *necessaria* (ma non sufficiente) per superare la prova.

Risposta

CALCOLA_CANTO_TELEPASS (u: codfis, m: INTEGER, a: INTEGER):

INSIEME(*ca*: STR, *cu*: STR, *i*: TIMESTAMP, *x*: TIMESTAMP, *d*: DENARO?)

Q = (SELECT *ca.NOME*, *cu.NOME*, *2.ISTAcc*, *2.ISTUsc*,
 $(\text{ABS}(\text{ca.DISTANZA} - \text{cu.DISTANZA}) \cdot t.\text{EUROALKM}) \text{ AS } d$

FROM CASELLO *ca*, CASELLO *cu*, ACCESSO *a*, VEICOLO *v*, TARIFFE *t*
WHERE *a.CONTELEPASS* = TRUE AND *a.USCITA* = TRUE AND *a.CLIENTE* = :u AND
ca.NOME = *a.CASACC* AND *cu.NOME* = *a.CASUsc* AND *a.VEICOLO* = *v.TARGA*
AND *v.CLASSE* = *t.CLASSE* AND *t.AUTOSTRADA* = *ca.AUTOSTRADA* AND
EXTRACT(MONTH FROM *ISTAcc*) = :m AND
EXTRACT(YEAR FROM *ISTAcc*) = :a)

RETURN *Q*

CALCOLA_COSTO_PEDAGGIO (d: STR, cu: STR): DENARO

ISERROR = (EXISTS (SELECT *
FROM ACCESSO *a'*, CASELLO *ca*, CASELLO *cu'*
WHERE *a'.COOTAGLIANO* = :d AND *a'.CASACC* = *ca.NOME*
AND *cu'*.NAME = *cu.NOME* AND (*a'.USCITA* = TRUE
OR *a'.CASUsc* = *cu* OR *ca.AUTOSTRADA* <? *cu.AUTOSTRADA*)))

IF *ISERROR*

GENERA ERRORE

Q = (SELECT ABS(*ca.DISTANZA* - *cu'.DISTANZA*) · *t.EUROALKM* AS *"x"*
FROM ACCESSO *a'*, CASELLO *ca*, CASELLO *cu*, TARIFFE *t*, VEICOLO *v*
WHERE *a'.COOTAGLIANO* = :d AND *a'.CASACC* = *ca.NOME*
AND *cu'.NAME* = *cu.NOME* AND *a'.VEICOLO* = *v.TARGA* AND
t.CLASSE = *v.CLASSE* AND *t.AUTOSTRADA* = *ca.AUTOSTRADA*)

RETURN *Q.x*

Risposta alla Domanda 8 (segue)

AVVIA TRACCIAMENTO (t: INTEGER, v: TARGA)

ISERROR = (EXISTS (SELECT * FROM TRACCIAMENTO t'
 WHERE $t'.$ VEICOLO = :v AND $t'.$ TUTORS = :t AND $t'.$ FINE IS NULL)
 OR NOT EXISTS (SELECT *
 FROM Accesso a, CANTIERE c, VEICOLO v', TUTOR t', LIVEL l
 WHERE $t'.$ ID = :t AND $v'.$ TARGA = :v AND
 a.USCITA = FALSE AND a.VEICOLO = :v AND
 a.CASACC = c.NOME AND a.ISTACC <= CURRENT_TIMESTAMP AND
 AND $t'.$ AUTOSTRADA = c.AUTOSTRADA AND v.CLASSE = l.CLASSE
 AND l.TUTORS = :t)))

IF ISERROR
 GENERA ERRORE

INSERT INTO TRACCIAMENTO (ID, VEICOLO, TUTORS, INIZIO, FINE)

VALUES
 (DEFAULT, :v, :t, CURRENT_TIMESTAMP, NULL)

TERMINA TRACCIAMENTO (t: INTEGER, v: TARGA)

ISERROR = (NOT EXISTS (SELECT * FROM TRACCIAMENTO t'
 WHERE $t'.$ VEICOLO = :v AND $t'.$ TUTORS = :t AND $t'.$ FINE IS NULL))
 OR NOT EXISTS (SELECT *
 FROM Accesso a, CANTIERE c, VEICOLO v', TUTOR t', LIVEL l
 WHERE $t'.$ ID = :t AND $v'.$ TARGA = :v AND
 a.USCITA = FALSE AND a.VEICOLO = :v AND
 a.CASACC = c.NOME AND a.ISTACC <= CURRENT_TIMESTAMP AND
 AND $t'.$ AUTOSTRADA = c.AUTOSTRADA AND v.CLASSE = l.CLASSE
 AND l.TUTORS = :t)))

IF ISERROR
 GENERA ERRORE

[CONTINUA MINUTA 33]

Tempo totale stimato per svolgere questa prova: 180 minuti (tempo totale concesso: 300 minuti).
 [Spazio per minute. Questa pagina non sarà valutata a meno che non sia puntata da pagine precedenti.]

```

Q = (SELECT CURRENT_TIMESTAMP AS ADESSO,
         l'.LIMITE AS l,
         (ABS(t'.DISTPRIMO.DAA - t'.DISTSECONDO.DAA) /
          (360 * EXTRACT(EPOCH FROM ADESSO - tr.INIZIO))) AS m
      FROM TRACCIAMENTO tr, CONNATUTORI t', LIVELI l', VEICOLO v'
     WHERE tr.TUTORIS = :t AND tr.VEICOLO = :v AND tr.FINE IS NULL
       AND t'.ID = :t AND l'.TUTORIS = :t AND v'.TARGA = :v
       AND l'.CLASSE = v'.CLASSE)
  
```

IF Q.m > Q.l

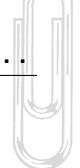
```

UPDATE TRACCIAMENTO tr
SET FINE = Q.ADESSO
WHERE tr.TUTORIS = :t AND tr.VEICOLO = :v AND tr.FINE IS NULL
  
```

ELSE

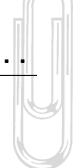
```

DELETE FROM TRACCIAMENTO
WHERE tr.TUTORIS = :t AND tr.VEICOLO = :v AND tr.FINE IS NULL
  
```



[Spazio per minute. Questa pagina non sarà valutata a meno che non sia puntata da pagine precedenti.]

[Spazio per minute. Questa pagina non sarà valutata a meno che non sia puntata da pagine precedenti.]



[Spazio per minute. Questa pagina non sarà valutata a meno che non sia puntata da pagine precedenti.]