



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.ER – Modulo risposte prova scritta

Dati dello studente e dell'esame

Cognome e nome: Matricola:

Data:

Corso di laurea e canale di appartenenza:

- Laurea in Informatica, canale 1 (A-L, Prof. G. Perelli)
- 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

CLIENTE

- Col: STRUTTURA
- DATA (SUPLIMENTO: DATE)

PRIVATO

- Nome
- Cognome
- CARTE: CARTA [1..*]

COMMERCIALE

- RAGIONE SOCIALE

SPECIFICA

- RISULTATO DA OTTENERE:
- RISULTATO A OTTENERE:
- INDIVIDUAZIONE DELL'OGGETTO:
- INDIVIDUAZIONE DELLA CONSEGUENZA:
- SISTEMA

CITTÀ

- Nome

NAZIONE

- Nome

CONTINENTE

- Nome

MERCE

- ASSICURAZIONI
- IMPRESA

SOGLIA PREZO

- PREZZO
- DA PREZO
- A PREZO

Depot

- ISIT ANNO:
- ISIT PREZO

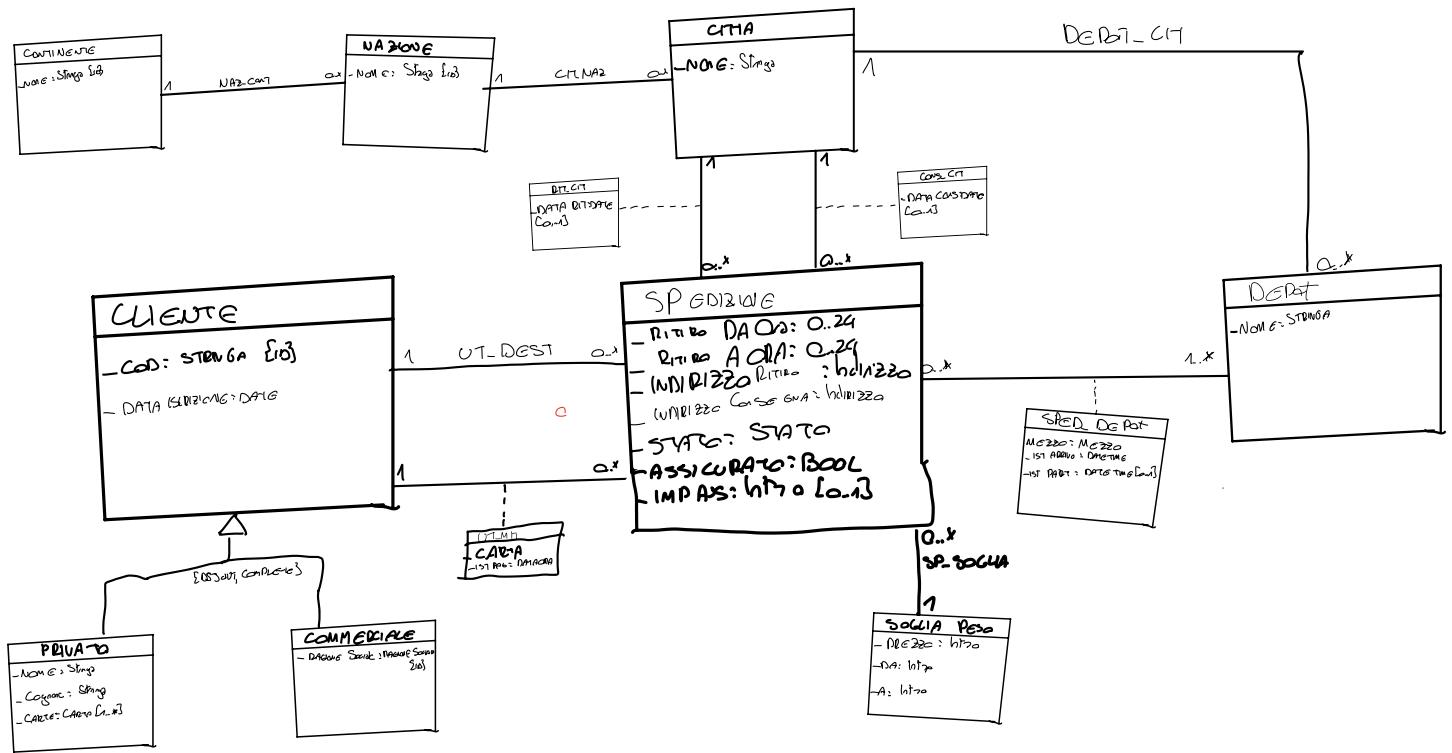
Risposta alla Domanda 1 (segue)

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

Produrre 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: ... <u>CLIENTE</u></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">attributo</th><th style="text-align: left;">dominio</th><th style="text-align: left;">moltep. (*)</th></tr> </thead> <tbody> <tr> <td></td><td></td><td></td></tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p><u>[V.CLIENTE. DATA_RITIRO DATA]</u></p> <p>$\forall c, s, d_1, d_2, dc$ $[CLIENTE(c) \wedge spedizione(s) \wedge utm(c_s) \wedge citta(c)] \rightarrow$ $[DATA_RITIRO(s, c_1, dc) \wedge DATA_RITIRO(s, c_2, dc)] \rightarrow d_1 = d_2$</p>	attributo	dominio	moltep. (*)				<p>2 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: ... <u>CLIENTE</u></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">attributo</th><th style="text-align: left;">dominio</th><th style="text-align: left;">moltep. (*)</th></tr> </thead> <tbody> <tr> <td></td><td></td><td></td></tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p><u>[V.CLIENTE. CONS DATA]</u></p> <p>$\forall c, s, d_1, dc$ $[CLIENTE(c) \wedge spedizione(s) \wedge utm(c_s) \wedge citta(c)] \rightarrow$ $[DATA_CONSEGNA(s, c_1, dc) \wedge DATA_CONSEGNA(s, c_2, dc)] \rightarrow d_1 = d_2$</p>	attributo	dominio	moltep. (*)			
attributo	dominio	moltep. (*)											
attributo	dominio	moltep. (*)											

<p>3 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: CLIENTE.....</p> <table border="1"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> </table>	attributo	dominio	moltep. (*)	<p>5 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: CLIENTE.....</p> <table border="1"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> </table>	attributo	dominio	moltep. (*)
attributo	dominio	moltep. (*)					
attributo	dominio	moltep. (*)					
<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>[V.CLIENTE.MTDEST]</p> <p>$\forall c, s$ $CLIENTE(c) \wedge SPENDERE(s) \wedge MT(c, s) \rightarrow MT_{DEST}(c, s)$</p>	<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>[V.CLIENTE.DESIMP]</p> <p>$\forall c, s$ $CLIENTE(c) \wedge SPENDERE(s) \wedge MT_{DEST}(c, s) \rightarrow MT_{MILL}(c, s)$</p>						

<p>4 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: PRIVATO.....</p> <table border="1"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> </table>	attributo	dominio	moltep. (*)	<p>6 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: PRIVATO.....</p> <table border="1"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> </table>	attributo	dominio	moltep. (*)
attributo	dominio	moltep. (*)					
attributo	dominio	moltep. (*)					
<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>[V.PRIVATO.CARTA PROPRIA]</p> <p>$\forall p, s, c, f$ $PRIVATO(p) \wedge Speciale(s) \wedge CARTA(p, c, f) \wedge$ $CARTA(p, c, f) \rightarrow c \in CP$</p>	<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>[V.PRIVATO.CARTA VALENTA]</p> <p>$\forall p, s, c, scad, ip$ $PRIVATO(p) \wedge Speciale(s) \wedge CARTA(p, s, f), scadenza(s, scad)$ $\wedge istpaG(p, s, ip) \rightarrow scad > ip$</p>						

<input checked="" type="checkbox"/> 7	Tipo: <u>Entità</u> Relationship (cerchiare)	
Nome:	<u>SPECIAZIONE</u>	
attributo	dominio	moltep. (*)

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

Vincoli:

[U. SPECIAZIONE. STATO]

$\forall s, c, dc, st$
 $[SPECIAZIONE(s) \wedge CITTA(c) \wedge DATA_RITIRO(s, c, dc) \wedge DATA_RITIRO(s, c, dc) = NULL \rightarrow \exists s1 \text{ STATO}(s, s1) \wedge \text{Nome}(s1, \text{Pendente})]$

<input checked="" type="checkbox"/> 9	Tipo: <u>Entità</u> Relationship (cerchiare)	
Nome:	<u>SPECIAZIONE</u>	
attributo	dominio	moltep. (*)

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

Vincoli:

[U. SPECIAZIONE. STATO]

$\forall s, c, dc, st, dc$
 $[SPECIAZIONE(s) \wedge CITTA(c) \wedge DATA_RITIRO(s, c, dc) \wedge DATA_RITIRO(s, c, dc) \neq NULL \wedge DATA_CORSO(s, c, dc) \wedge DATA_CORSO(s, c, dc) \neq NULL \rightarrow \exists s1 \text{ STATO}(s, s1) \wedge \text{Nome}(s1, \text{IN TRASITO})]$

<input checked="" type="checkbox"/> 8	Tipo: <u>Entità</u> Relationship (cerchiare)	
Nome:	<u>SPECIAZIONE</u>	
attributo	dominio	moltep. (*)

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

Vincoli:

[U. SPECIAZIONE. STATO]

$\forall s, c, dc, dc$
 $[SPECIAZIONE(s) \wedge CITTA(c) \wedge DATA_RITIRO(s, c, dc) \wedge DATA_RITIRO(s, c, dc) \neq NULL \wedge DATA_CORSO(s, c, dc) \wedge DATA_CORSO(s, c, dc) \neq NULL \rightarrow \exists s1 \text{ STATO}(s, s1) \wedge \text{Nome}(s1, \text{consegnata})]$

<input checked="" type="checkbox"/> 10	Tipo: <u>Entità</u> Relationship (cerchiare)	
Nome:	<u>SPECIAZIONE</u>	
attributo	dominio	moltep. (*)

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

Vincoli:

[U. SPECIAZIONE. DATE]

$\forall s, c, dc, dc$
 $[SPECIAZIONE(c) \wedge CITTA(c) \wedge DATA_RITIRO(s, c, dc) \wedge DATA_RITIRO(s, c, dc) \neq NULL \wedge DATA_CORSO(s, c, dc) \wedge DATA_CORSO(s, c, dc) \neq NULL \rightarrow DC \leq DR]$

<p>11 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: SPEDIZIONE.....</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">attributo</th><th style="text-align: left;">dominio</th><th style="text-align: left;">moltep. (*)</th></tr> </thead> </table>	attributo	dominio	moltep. (*)	<p>13 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: SPEDIZIONE.....</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">attributo</th><th style="text-align: left;">dominio</th><th style="text-align: left;">moltep. (*)</th></tr> </thead> </table>	attributo	dominio	moltep. (*)
attributo	dominio	moltep. (*)					
attributo	dominio	moltep. (*)					
<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>[V. SPEDIZIONE_FASCIA]</p> <p>$\forall s, d, a$ $[Spedizione(s) \wedge DataOra(s, d) \wedge DataAra(s, a)] \rightarrow DA \leq AA$</p>	<p>No SPED ALLO STESSO WO DELLA STessa CITTÀ NELLA STESSA NRZ</p> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>[V. SPEDIZIONE_WNRZ]</p> <p>$\forall s, d, c_1, c_2$ $[Spedizione(s) \wedge DataOra(s, d) \wedge DataAra(s, c_1) \wedge DataAra(s, c_2) \wedge c_1 = c_2] \rightarrow DA \neq DC$</p>						

<p>12 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: DEPOT.....</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">attributo</th><th style="text-align: left;">dominio</th><th style="text-align: left;">moltep. (*)</th></tr> </thead> </table>	attributo	dominio	moltep. (*)	<p>14 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: DEPOT.....</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">attributo</th><th style="text-align: left;">dominio</th><th style="text-align: left;">moltep. (*)</th></tr> </thead> </table>	attributo	dominio	moltep. (*)
attributo	dominio	moltep. (*)					
attributo	dominio	moltep. (*)					
<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>[V. DEPOT_ISSTANZA]</p> <p>$\forall D, A, I, P, S, C, D2, DC, C2$ $[Depot(D) \wedge Spedizione(S) \wedge DataOra(D, S, I) \wedge DataOra(D, S, P) \wedge DataOra(D, S, C) \wedge DataOra(D, S, DC) \wedge DataOra(D, C, DC) \wedge DataOra(C, DC, DC)] \rightarrow DC \neq IA \wedge DC \neq IC$</p>	<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>[V. DEPOT_ISSTANZA]</p> <p>$\forall D, A, I, P, S$ $[Depot(D) \wedge Spedizione(S) \wedge IsstAdrivo(D, S, I) \wedge IsstParr(D, S, P) \wedge IsstParr(D, S, I) \wedge IsstParr(D, S, P)] \rightarrow IA \neq IP$</p>						

15	Tipo: Entità Relationship (cerchiare)
	<i>MOLTE</i>
Nome:

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

Vincoli:

[V. MOLTE ASSICURATA]

$\forall s, a_1, s_1$

$\{ \text{Spese}(\text{S}) \wedge \text{ASSICURATA}(s, a, \text{"FALSE"}) \wedge \text{IMPOSTA}(s, 1) \} \rightarrow \text{HNBK}$

17	Tipo: Entità Relationship (cerchiare)
	<i>MOLTE</i>
Nome:

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

Vincoli:

[V. MOLTE ASSICURATA]

$\forall s, a_1, s_1$

$\{ \text{Spese}(\text{S}) \wedge \text{ASSICURATA}(s, a, \text{"TRUE"}) \wedge \text{IMPOSTA}(s, 1) \} \rightarrow \text{HNBK}$

16	Tipo: Entità Relationship (cerchiare)
	<i>SOGIA PESO</i>
Nome:

attributo dominio moltep. (*)

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

Vincoli:

[V. SOGIA PESO DA A]

$\forall s, p, a$

$\{ \text{SogiaPeso}(\text{S}) \wedge \text{PA}(\text{S}, p) \wedge \text{A}(\text{S}, a) \} \rightarrow \text{DA}$

18	Tipo: Entità Relationship (cerchiare)

Nome:

attributo dominio moltep. (*)

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

Vincoli:

17 Vincoli

ESTERNO //

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

TIPI DI DATI:

CARTA = $\{ \text{NOMINA}: \text{STRNG}, \text{NUMERO}: \text{INT70}, \text{DATA SCADENZA} = \text{DATE} \}$

STATO enum = $\{ \text{"PENDENTE"}, \text{"IN TRANSITO"}, \text{"CONSEGNATO"} \}$

MEZZO enum = $\{ \text{"Autocarro"}, \text{"NAVE"}, \text{"Aereo CAP60"} \}$

INDIRIZZO = $\{ \text{VIA}: \text{STRNG}, \text{N_CIV} = \text{INT70}, \text{CAP} = \text{CAP} \}$

CAP STRNGA secondo REGEX

RAGLIO E SOCIALE STRNGA secondo REGEX

OPERAZIONI DI CLASSE

CLASSE SPEDIZIONE

PREZZO () : Prezzo

PRE: NESSUNA

PST: Non modifica il liv. ESTENSIONALE

$$M_{1n} = M_{2n}$$

Sia s l'oggetto del mondo che chiama la funzione

$$PP = \left(P \right)_{\substack{\text{sulla Poco}(\text{sp}) \wedge \text{sp} - \text{so} \text{ guidato}, \text{sp} \wedge \\ \text{prezzo}(\text{sp}, p) }}$$

$$SIA IS = \left(I \right)_{\substack{\text{Assicurato}(\text{this}, p), \text{"true"} \wedge \text{IMPASS}(\text{this}, 1) }}$$

SE IS != NIL

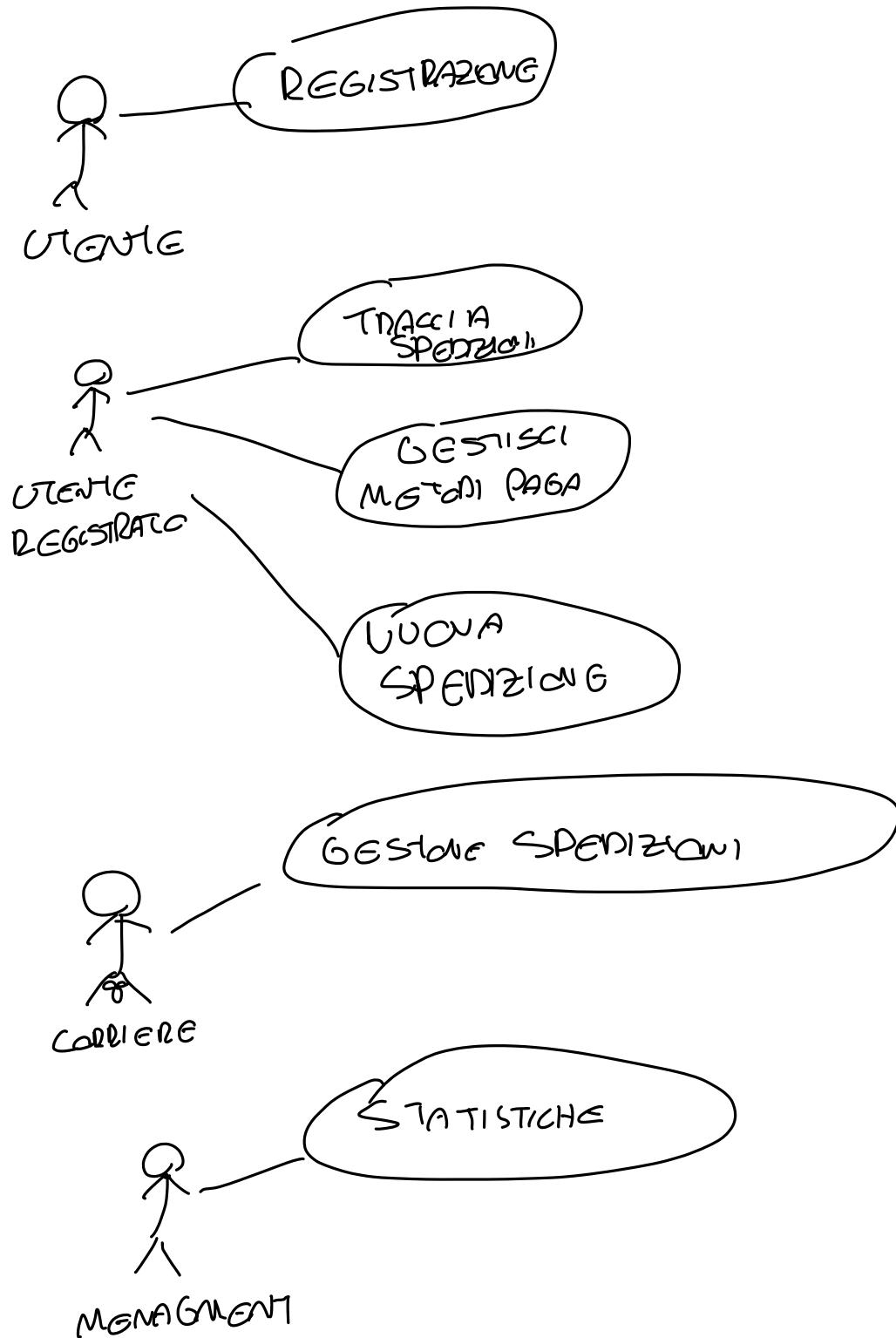
$$\text{RESULT} = PP \cdot (\text{co1} \cdot IS)$$

ALTRIMENTI

$$\text{RESULT} = PP$$

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: REGISTRAZIONE (nome use-case)

Operazioni dello use-case:

REGISTRAZIONE_PRIVATO (NOME: STRINGA) UTENTE Privato

REGISTRAZIONE_COMMERCIAL (RG: RAGONE SOCIALE): Utente Commerciale

2 Specifica use-case: TRACCIASIPOZIONE (nome use-case)

Operazioni dello use-case:

TRACCIA_SPOZIONE (S: SPEDIZIONE)

3 Specifica use-case: GESTISCI_MECOI_PAGAMENTO (nome use-case)

Operazioni dello use-case:

GESTISCI_PAG (N: STRINGA, NU: INTERO, D: DATA); CARTA

Specifica use-case: Nucleo SPEDIZIONE (nome use-case)

Operazioni dello use-case:

SPEDIZIONE (RD: Intero, RA: Intero, R: Intero, C: Intero, D: Città, CC: Città) : SPEDIZIONE

Specifica use-case: GESTIONE SPEDIZIONI (nome use-case)

Operazioni dello use-case:

PRENDI_IN_CARICO (s: Spedizione)

COMPLETA (s: Spedizione)

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

Operazioni dello use-case:

STATISTICHE (c1: Contingente, c2: Contingente : DI: DATA, DF: DATA) : REALE

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

STATISTICS (c_1 : Contratto, c_2 : Contante; DI: DATA, DF: DATA) : REALE

PRE : DI \sqsubseteq DF

POST:

$$S = \{ s \mid \begin{array}{l} \text{SPEDIZIONE}(s) \wedge \exists m, n_1 \quad \text{CITA}(c_1) \wedge \text{DITC}(s, c_1) \\ \text{NAZIONE}(n_1) \wedge \text{CIT NAZ}(c_1, n_1) \wedge \text{NAZ-CNT}(m, c_1) \wedge \\ \exists c_2, n_2 \quad \text{CITA}(c_2) \wedge \text{CONS-CIT}(s, c_2) \wedge \text{NAZIONE}(n_2) \wedge \\ \text{CIT NAZ}(c_2, n_2) \wedge \text{NAZ-CNT}(n_2, c_2) \end{array} \}$$

$$\text{Tot} = \sum_{s \in S} \text{prezzi}(s)$$

$$\text{RESULT} = \frac{\text{Tot}}{|S|}$$

TRACCIA Spedizione (s : Specifonic) : $\{ [\text{Depot}, \text{Mezza}, \text{Start}, \text{Stop}] \}_{\text{out}} \text{STATO}_{c_1, s} \}$

PRE : Nessuna

D-TC

$$\text{SIA } DM = \{ (D, M, I, P) \mid \exists D, M, I, P \quad \text{DEPOT}(D) \wedge \text{SPED-DEP}(s, D) \wedge \text{MEZZA}(s, M) \\ \wedge \text{START}(s, D, I) \wedge \text{STOP}(s, D, P) \}$$

$$\text{SIA } \text{STATO} = \text{STATO}(s)$$

$$DM = \text{Ord}(DM, \text{Depot-ord})$$

SE STATO = NULL :

$$\text{RESULT} = DMC$$

$$\text{ALTRIMENTI } \text{RESULT} = (DM, \text{STATO})$$

[continua alla pagina seguente]

Risposta alla Domanda 5 (segue)

Dopo $\sqsubset \sqsubset ((D_1, M_1, A_1, P_1), (D_2, M_2, A_2, P_2))$: B_{20}

PRE: Nessuna

POST:

Se I_1, I_2 (DATA) tali da soddisfare la formula segnata;

$\exists SD_1_066, SD_2_066, IS_1_066, IS_2_066$

$\text{SPED_DEPOT}(S, D_1, SD_1_066) \wedge \text{IS1ADDVO}(SD_1_066, IS_1_066)$

$\wedge \text{VALORE}(IS_1_066, I_1) \wedge$

$\text{SPED_DEPOT}(S, D_2, SD_2_066) \wedge \text{IS1ADDVO}(SD_2_066, IS_2_066)$

$\wedge \text{VALORE}(IS_2_066, I_2)$

RESULT = TRUE $\Leftarrow (I_1 < I_2)$

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 String >= varchar(255) check (value not null and value ~a)
 Create Domain Integer >= integer check (value not null and value >= 0)

 Create Domain Integer ?c >= integer check (value not null and value >= 0)
 Create Type INDIANZO = { v1: String, v2: Integer, v3: Integer }

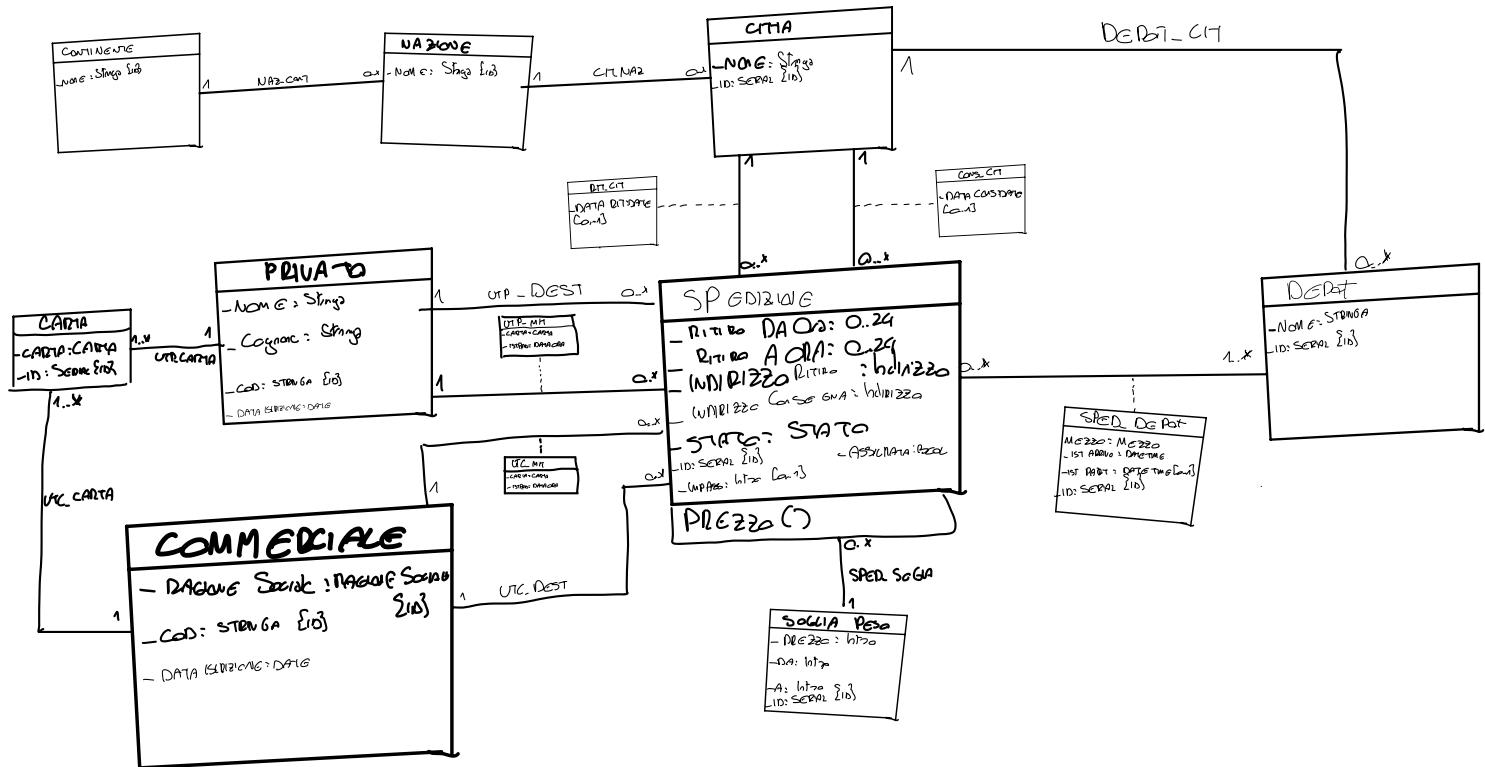
 CREATE Domain RazoneSociale >= varchar(255) check (value not null and value ~a)

 Create Type CAPTA = { mlnativo: String, numero: Integer, dataSociazione: DATE }

 CREATE Type Stato AS ENUM = { "PENDENTE", "IN TRANSITO", "CONSEGNATO" }

 CREATE TYPE MEZZO AS ENUM = { "AUTOCARRO", "NAVE", "AEREO CARBO" }

Diagramma ER ristrutturato



Breve descrizione delle scelte effettuate durante la ristrutturazione

DIVISIONE DELLA GENERALIZZAZIONE SU CLIENTE PER AVER
2 Tabelle Client

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)

Vincoli a GOUVATI

[V. PRIVATO. NO COM]

$$\forall u \quad \text{Privato}(u) \rightarrow \neg \text{Commerciale}(u)$$

[V. COMMERCIALE. NO PRIV]

$$\forall u \quad \text{Commerciale}(u) \rightarrow \neg \text{Privato}(u)$$

Vincoli MODIFICA-II

[V. CLIENTE. MIT DEST]

[V. CLIENTE. DEST MIT]



[V. PRIVATO. MIT DEST]

$\forall c, s$

$$\text{PRIVATO}(c) \wedge \text{SPEDIZIONE}(s) \wedge \text{UT_MIT}(c, s) \rightarrow \exists \text{OT_DEST}(c, s)$$

[V. PRIVATO. DEST MIT]

$\forall c, s$

$$\text{PRIVATO}(c) \wedge \text{SPEDIZIONE}(s) \wedge \text{UT_DEST}(c, s) \rightarrow \exists \text{OT_MIT}(c, s)$$

[V. COMMERCIALE MIT DEST]

$\forall c, s$

$$\text{Commerciale}(c) \wedge \text{SPEDIZIONE}(s) \wedge \text{UT_MIT}(c, s) \rightarrow \exists \text{OT_DEST}(c, s)$$

[V. COMMERCIALE. DEST MIT]

$\forall c, s$

$$\text{Commerciale}(c) \wedge \text{SPEDIZIONE}(s) \wedge \text{UT_DEST}(c, s) \rightarrow \exists \text{OT_MIT}(c, s)$$

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 CONTINENTE... (nome)	Derivante da: entità relationship (cerchiare)
---	--

Attributi <u>Nome</u>								
-------------------------	--	--	--	--	--	--	--	--

Domini <u>STRINGA</u>								
-------------------------	--	--	--	--	--	--	--	--

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:

2 Relazione NAZIONE..... (nome)	Derivante da: entità relationship (cerchiare)
--	--

Attributi <u>Nome</u>	<u>CONTINENTE</u>							
-------------------------	-------------------	--	--	--	--	--	--	--

Domini <u>STRINGA</u>	<u>STRINGA</u>							
-------------------------	----------------	--	--	--	--	--	--	--

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: CONTINENTE REFERENCES CONTINENTE (Nome)
unique (Nome,CONTINENTE)

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

3 Relazione CITTÀ..... (nome)	Derivante da: entità relationship (cerchiare)
--------------------------------------	--

Attributi <u>ID</u>	<u>Nome</u>	<u>NAZIONE</u>						
-----------------------	-------------	----------------	--	--	--	--	--	--

Domini <u>SERIAL</u>	<u>STRINGA</u>	<u>STRINGA</u>						
------------------------	----------------	----------------	--	--	--	--	--	--

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: NAZIONE REFERENCES NAZIONE (Nome)
unique (ID,NAZIONE)

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

4 Relazione PROVATO..... (nome)	Derivante da: entità relationship (cerchiare)
--	--

Attributi <u>CAN</u>	<u>NOME</u>	<u>COLONNE</u>	<u>DATI</u> <u>STRINGS</u>					
------------------------	-------------	----------------	----------------------------	--	--	--	--	--

Domini <u>STRINGA</u>	<u>STRINGA</u>	<u>STRINGA</u>	<u>NOME</u>					
-------------------------	----------------	----------------	-------------	--	--	--	--	--

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):

[v. inclusione] PROVATO occorre in VIP_CARTA

La relazione accorda le relazioni che implementano le seguenti relationship:

5 Relazione VIP_CARTA... (nome)	Derivante da: entità relationship (cerchiare)
--	--

Attributi <u>PROVATO</u>	<u>CARTA</u>							
----------------------------	--------------	--	--	--	--	--	--	--

Domini <u>STRINGA</u>	<u>Integ.</u>							
-------------------------	---------------	--	--	--	--	--	--	--

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: PROVATO references PROVATO (CAN)
fk: CARTA references CARTA (Integ.)

La relazione accorda le relazioni che implementano le seguenti relationship:

6 Relazione ...CARTA..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> CARTA	
Domini SERIAL CARTA	

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:

7 Relazione COMMERCIALE..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>CDN</u> DRAgone Socia NOME SOCIANO	
Domini STRING <u>rendone sociale</u> DATE	

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):

UNI (AGIO & SOCIE)

[U. INCLUSIOn] COMMERCIALE OCCORRE IN VTC-CARTE

La relazione accorda le relazioni che implementano le seguenti relationship:

8 Relazione ...VTC-CARTE..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi COMMERCIALE <u>CARTA</u>	
Domini STRING 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: COMMERCIALE references COMMERCIALE

fk: CARTA references CARTA (ii)

La relazione accorda le relazioni che implementano le seguenti relationship:

9 Relazione SPEDIZIONE.... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> DATA_RITIRO DATA_PRESA ID_CARS STATO DEST_PDO * DEST_COM *	
Domini SERIAL INT64 INT64 INT64 STATO STRING STRING	

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):

check DATA_RITIRO <= DATA_PRESA

fk: SOLO_PESO references SOLO_PESO (ii)



La relazione accorda le relazioni che implementano le seguenti relationship: ...VTC-DEST, ...VTC-PDO, ...SPED-SOGNA

10 Relazione ...UTP_MIT..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>PRIVATO</u> <u>SPEDIZIONE</u> CARTA IST_RR6	
Domini STRING INTEGER INTEGER DATE-TIME	

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: PRIVATO REFERENCES PRIVATO (ii)

fk: SPEDIZIONE REFERENCES SPEDIZIONE (ii)

fk: CARTA REFERENCES CARTA (ii)

La relazione accorda le relazioni che implementano le seguenti relationship:

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

Attributi	<u>COMMERCIALS</u>	<u>SPEDIZIONE</u>	<u>CARICA</u>	IST_R_6				
Domini	STADROP	INTEGER	INTEGER	DATETIME				

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: COMMERCIAL REFERENCES COMMERCIAL (con)

fk: SPEDIZIONE REFERENCES SPEDIZIONE (in)

fk: CARICA REFERENCES CARICA (in)

La relazione accorda le relazioni che implementano le seguenti relationship:

12 Relazione Sp.GIA. PESSO..... (nome) Derivante da: entità | relationship (cerchiare)

Attributi	<u>ID</u>	MEZZO_E:	DATA_E:	ATA_E:				
Domini	Serial	Intero	Intero	Intero				

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):

CHECK DATA_E >= ATA_E

La relazione accorda le relazioni che implementano le seguenti relationship:

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

Attributi	<u>ID</u>	None	CITTA					
Domini	Serial	STADROP	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: CITTA REFERENCES CITTA (in)

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

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

Attributi	<u>SPEDIZIONE</u>	<u>DEPOT</u>	<u>IN</u>	<u>MESZZO</u>	IST_PARSO	IST_PARI		
Domini	INTEGER	INTEGER	Serial	MESZZO	DATETIME	DATETIME		

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):

Unique (in)

CHECK IST_ADDRESS & IST_PHONE

La relazione accorda le relazioni che implementano le seguenti relationship:

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

Attributi	<u>SPEDIZIONE</u>	<u>CITTA</u>	DITRA_DITRA					
Domini	INTEGER	INTEGER	DATE					

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: SPEDIZIONE REFERENCES SPEDIZIONE (in)

fk: CITTA REFERENCES CITTA (in)

La relazione accorda le relazioni che implementano le seguenti relationship:

16 Relazione <u>CORS-CITI</u> (nome)	Derivante da: entità relationship (cerchiare)						
Attributi <u>SPEDIZIONE</u> <u>CITTA</u> DATA_CORS *							
Domini <u>Intero</u> <u>Intero</u> <u>Data</u>							

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: SPEDIZIONE REFERENCES SPEDIZIONE (n)
fk: CITTA REFERENCES CITTA (n)

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 ennupla); (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.

TRIGGERS DA IMPLEMENTARE

- [T. PRIVATO. NO COMM]
- [T. SPEDIZIONE. ASSICURATA] → SE sped → ASSICURATA IMPASS+NULL
- [T. SPEDIZIONE. INDIRIZZO] → NEL CASO CHE SE RTTC HA IL SISTEMA PROTEGGE

T. PRIVATO. NO COMM

INSEGNAMENTO SU PRIVATO

PRE OPERAZIONE:

```
if VALID:=NOT exist ( SELECT P.ID
    FROM PRIVATO P, COMMERCIALE C
    WHERE P.ID=C.ID AND NEW.ID=P.ID )
```

if VALID:
COMMIT

else:
GENERA ERRORE
REVERT

T. SPEDIZIONE. ASSICURATA

INSEGNAMENTO IN SPEDIZIONE

PRE OPERAZIONE:

```
if VALID:=NOT exist ( SELECT S.ID
    FROM SPEDIZIONE S
    WHERE S.ASSICURATA=TRUE AND NEW.ID=S.ID AND
    S.IMPASSIVEL )
```

if VALID:
COMMIT

else:
GENERA ERRORE
REVERT

T. SPEDIZIONE. INDIRIZZO

INSEGNAMENTO IN SPEDIZIONE

PRE OPERAZIONE:

```
if VALID:=NOT exist ( SELECT S.ID
    FROM SPEDIZIONE S, RTT-CIT RC, CITTÀ C1, CITTÀ C2
    WHERE NEW.ID=S.ID AND S.ID=RC.SPEDIZIONE AND RC.CITTA=C1.ID AND C1.SPEDIZIONE=S.ID
    AND RC.CITTA=C2.ID AND C1.ID=C2.ID )
```

if VALID:
COMMIT

else:
GENERA 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

OPERAZIONI CLASSE

CLASSE SPEDIZIONE

- PREZZO (S: Integer): DEC6

CREATE FUNCTION

Q = (SELECT SP.PREZZO
 FROM SPEDIZIONE S, SOGLIA_PESO SP
 WHERE S.ID = S AND S.SOGNAPESO = SP.ID)

return Q

USE CASE

STATISTICS(C1: STANZA, C2: STANZA, D1: DATE, D2: DATE): DEC6

IMPLEMENTAZIONE IN Q LA SEGUENTE QUERY SQL

SELECT C1(SUM(PREZZO(S.ID))/COUNT(DISTINCT S.ID))
 FROM SPEDIZIONE S, RT_C1 RC, CONS_C1 CC
 LEFT JOIN NAZIONE N1 ON CC.NAZIONE = N1.NOME
 LEFT JOIN NAZIONE N2 ON CC.NAZIONE = N2.NOME
 WHERE S.ID = RC.SPEDIZIONE = CC.SPEDIZIONE
 AND RC.CITTA = C1 AND CC.CITTA = C2
 AND C1.NAZIONE = N1.NOME AND CC.NAZIONE = N2.NOME
 AND N1.CONTINENTE = C1 AND N2.CONTINENTE = C2

RESULT = Q

Risposta alla Domanda 8 (segue)

TRACCIA Spedizione (S: hTegy) = $\{ \{ \text{DEPOT}, \text{MEZZA}, \text{START}, \text{STOP} \} [l..x], \text{START}[c..1] \}$

MEMORIZZA IN Q IL RISULTATO DELLA SEQUENZA RUGBY SQL

SELECT D.NOME, SD.MEZZA, SD.START, SD.STOP

FROM DEPOT D, SPEDIZIONE S, SPEDIZIONE SD

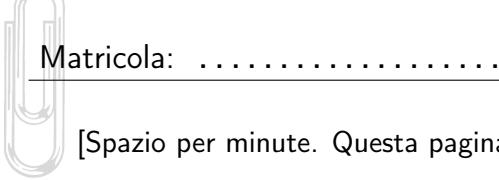
WHERE D.ID=SD.DEPOT AND S.ID=SD AND SD.SPEDIZIONE=S.ID

ORDER BY SD.START ASC

RESULT = Q

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.]

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



Matricola:

Minute

[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.]