

Nome:	Cognome:	Matricola:

Esercizio 1

Si consideri la seguente tabella contenente le informazioni che descrivono l'attività di una compagnia di assicurazioni.

Insurance (CodCliente, ResidenzaCliente, Veicolo, ClasseVeicolo, Abitazione, IndirizzoAbitazione, Pensione, IncidentiCasa, Furto, PremioAnnuoVeicolo, PremioAnnuoFurto, PremioAnnuoCasa, PremioAnnuoPensione).

Ogni cliente può avere assicurazioni per differenti auto con premi diversi, ma una sola pensione integrativa.

Ogni assicurazione ha un solo intestario.

L'assicurazione sul furto può riguardare un'auto o una casa; l'assicurazione sugli incidenti in casa riguarda una sola abitazione; l'abitazione assicurata deve essere di proprietà del cliente. Un cliente può possedere più abitazioni.

a) Individuare la chiave e tutte le dipendenze funzionali non banali

Un cliente esiste nel Database solo se ha almeno una assicurazione attiva.

Un cliente potrebbe non avere tutte le assicurazioni possibili.

IncCasa → PACasa, Abitazione, CodCliente, IndAbitazione

Abitazione, Furto → PFurto

Veicolo, Furto → PFurto

CodCliente → Residenza

Veicolo → ClasseVeicolo, Pveicolo, CodCliente

Pensione → PPensione, CodCliente

Chiave: IncCasa, Furto, Veicolo, Pensione

N.B. Questa non è una chiave primaria, ma almeno un attributo non è mai nullo.

Nel caso sia noto che un cliente ha sottoscritto tutte le assicurazioni fornite dalla compagnia

IncCasa → PACasa

Abitazione → CodCliente, IndAbitazione, Furto, IncCasa

Furto → PFurto

CodCliente → Residenza, Pensione

Veicolo → ClasseVeicolo, PVeicolo, Furto, CodCliente

Pensione → PPensione

Chiave: Veicolo, Abitazione

b) Verificare se Insurance è in terza forma normale e, se non lo è, portarla in terza forma normale.

Insurance non è in terza forma normale, per portarla in terza forma normale si fa una tabella per dipendenza e, in entrambi i casi, si aggiunge una tabella contenete la chiave di Insurance.

Esercizio 2

Date le relazioni $R(A,B,C)$, $S(D,E)$, scrivere l'interrogazione ottimizzata in algebra relazionale corrispondente alla seguente interrogazione SQL.

```
select R1.A, S.E
from R R1, R R2, S
where R1.C = R2.C
and R1.B = S.D
and R1.A > 2 and S.E > R2.A.
```

$$\Pi_{A,E} (\Pi_{A1,B1,A2} (\rho_{R1 \leftarrow R} (\sigma_{A > 2 = CP} (R)) \triangleright \triangleleft_{C1=C2} \rho_{R2 \leftarrow R} (\Pi_{A,C} (R))) \triangleright \triangleleft_{B1=D \wedge E > A2} S)$$

Esercizio 3

Considerare la seguente base di dati:

- **Aeroporto** (Città, Nazione, Continente)
- **Volo** (CodVolo, TipoAereo, GiornoSettimana, CittàPartenza, OraPartenza, CittàArrivo, OraArrivo, CodCompagnia, NumScali, LowNormal)
- **Aereo** (TipoAereo, NumPasseggeri, QuantMerchi)
- **Scali** (CodVolo, GiornoSettimana, Città, OraArrivo, OraPartenza)

- a) Scrivere una espressione in algebra relazionale che elenchi i voli, i giorni e le città di partenza in cui per andare a Melbourne provenienti da qualunque aeroporto europeo ci si ferma a Dubai per più di 4 ore.

$$\Pi_{CV,CP,GS} (\Pi_{Città} (\sigma_{C='Europa'} (Aeroporto)) \triangleright \triangleleft_{Città=CP} \Pi_{CV,CP,GS} (\sigma_{NS \geq 1 \wedge CA='Melbourne'} (Volo) \triangleright \triangleleft_{\sigma_{Città='Dubai'} \wedge OP-OA > 4} (Scali)))$$

Esercizio 4

Verificare se esiste e qual è uno schedule seriale conflict equivalente al seguente.

$r_1(x) \ r_2(x) \ w_1(x) \ r_2(y) \ w_1(y) \ w_2(z) \ r_3(z) \ r_1(z) \ w_3(x) \ w_1(z) \ w_3(z) \ r_3(y) \ w_3(y)$

Il grafo dei conflitti di questo schedule causa un ciclo tra la Transazione 1 e la 3, quindi non esiste nessun schedule seriale conflict equivalente.