

Esame di Tecnologia delle basi di dati

28 Aprile 2005

1. (13 punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate):

```
LINEA(CodL, LuogoPartenza, LuogoArrivo, Distanza)
VOLO(CodL, Data, OraPartenza, OraArrivo, CodA)
AEREO(CodA, MarcaA, ModelloA, NumPosti)
DIPENDENTE(CodFiscDip, NomeDip, CognomeDip, Qualifica, DataNascita)
ABILITATO(CodFiscDip, CodA)
TURNO(CodFiscDip, Data, OraInizio, OraFine)
```

Si ipotizzino le seguenti cardinalità per le tabelle:

- $\text{card}(\text{LINEA}) = 10^4$ tuple
- $\text{card}(\text{VOLO}) = 10^7$ tuple per l'anno 2004
- $\text{card}(\text{AEREO}) = 10^2$ tuple
- $\text{card}(\text{DIPENDENTE}) = 10^5$ tuple
- $\text{card}(\text{ABILITATO}) = 10^4$ tuple
- $\text{card}(\text{TURNO}) = 10^8$ tuple per l'anno 2004

Si considerino le seguenti query SQL:

- (a)

```
select L.CodL, V.Data, V.OraPartenza, V.OraArrivo, AE.Marca
from LINEA L, VOLO V, AEREO AE
where V.CodA=AE.CodA and L.CodL=V.CodL
      and L.LuogoPartenza='Roma' and L.LuogoArrivo='New York'
      and V.Data ≥ 1/03/04 and V.Data ≤ 31/03/04
      and AE.Marca='Boeing'
      and exists (select *
                  from DIPENDENTE D, ABILITATO AB
                  where D.Qualifica='Pilota' and D.CodFiscDip=AB.CodFiscDip
                  and AE.CodA=AB.CodA)
```
- (b)

```
select D.CodFiscDip, D.NomeDip, D.CognomeDip
from DIPENDENTE D, ABILITATO AB
where D.CodFiscDip=AB.CodFiscDip and D.Qualifica='Pilota'
      and D.DataNascita ≥ 1/1/1957 and
      not exists (select T.CodFiscDip
                  from TURNO T
                  where T.Data ≥ 1/08/04 and T.Data ≤ 31/08/04
                  and T.CodFiscDip=D.CodFiscDip
                  group by T.CodFiscDip
                  having count(*) ≥ 7)
group by D.CodFiscDip, D.NomeDip, D.CognomeDip
having count(*) ≥ 2
```

Separatamente per ciascuna delle due interrogazioni SQL

- (a) Si scriva l'espressione algebrica corrispondente.

- (b) Si scriva il piano di esecuzione che potrebbe essere scelto dall'ottimizzatore in assenza di strutture fisiche per migliorare le prestazioni dell'interrogazione.
- (c) Si scelgano una o più strutture fisiche accessorie per migliorare le prestazioni dell'interrogazione. Si motivi la scelta e si definisca il nuovo piano di esecuzione ipotizzato.

NOTA BENE: Sia per il punto b) che per il punto c)

- dove necessario, si ipotizzi la distribuzione dei dati.
- nel descrivere il piano di esecuzione si indichi quali operazioni sono svolte, in che ordine e, dove possibile, quale tipo di operatore è usato per svolgere una certa operazione (ad esempio in caso di join).

2. (13 Punti) Sono date le relazioni seguenti:

```
RICHIESTA-NOLEGGIO(CodRich, DataRich, CodSocio, DataInizioNoleggio,
DataFineNoleggio, Modello)
RINUNCIA-NOLEGGIO(CodRin, DataRin, CodSocio, DataInizioNoleggio, CodB)
NOLEGGIO(CodB, DataInizio, CodSocio, DataFine)
NOLEGGIO-IN-ATTESA(CodRich, DataRich, CodSocio, DataInizioNoleggio,
DataFineNoleggio, Modello)
BARCA(CodB, NomeB, Modello, Marca, AnnoFabbr)
```

Si scrivano i trigger che gestiscono una richiesta di noleggio e una rinuncia al noleggio. Per la richiesta di noleggio, occorre verificare se esiste una barca del modello prescelto non in prestito nel periodo richiesto. In caso affermativo, si definisce un nuovo noleggio nella tabella NOLEGGIO. Nel caso in cui più barche siano disponibili, si noleggia quella di fabbricazione più recente (per semplicità si supponga che ve ne sia sempre una sola). Se invece nessuna barca è disponibile nel periodo richiesto, si inserisce la richiesta tra le richieste in attesa di essere processate (tabella NOLEGGIO-IN-ATTESA).

Nel caso di rinuncia a un noleggio, si deve eliminare il noleggio dalla tabella NOLEGGIO. Si deve inoltre verificare se esistono richieste di noleggio in attesa nel periodo indicato e per il modello reso libero. In caso affermativo, si sceglie la richiesta in attesa da più tempo (per semplicità si supponga che ve ne sia sempre una sola), si definisce un nuovo noleggio per quella barca e si elimina la richiesta in attesa dalla tabella NOLEGGIO-IN-ATTESA.

3. (4 Punti) Si consideri il seguente schedule:

$r_1(B)r_4(A)r_3(C)w_2(C)w_4(A)w_1(B)r_3(A)r_1(A)r_2(B)w_3(A)$

Indicare se lo schedule è VSR o CSR serializzabile. Se serializzabile, indicare un'esecuzione seriale corrispondente.

Esame di Tecnologia delle basi di dati

11 Luglio 2005

1. (13 punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate):

ATTIVITA(IdA, NomeA, Livello)
LEZIONI(IdL, IdA, CodFiscIstr, Data, OraInizio, NumIscritti, Costo, Durata)
SOCI(NumTessera, CodFiscS, NomeS, CognomeS, DataNascita, DataT, ScadenzaT)
ISCRIZIONI-LEZIONI(NumTessera, IdL, DataIscr, Acconto)

Si ipotizzino le seguenti cardinalità per le tabelle:

- $\text{card}(\text{ATTIVITA}) = 10^2$ tuple
- $\text{card}(\text{LEZIONI}) = 10^6$ tuple per l'anno 2004
- $\text{card}(\text{SOCI}) = 10^5$ tuple
- $\text{card}(\text{ISCRIZIONI-LEZIONI}) = 10^8$ tuple per l'anno 2004

Si considerino le seguenti query SQL:

- (a)

```
select S.NomeS, S.CognomeS
from SOCI S, ISCRIZIONI-LEZIONI IL
where S.NumTessera=IL.NumTessera
      and S.DataNascita = 1980 and
      not exists (select *
                  from LEZIONI L
                  where IL.IdL=L.IdL and L.Durata ≤ 90)
group by S.NumTessera, S.NomeS,S.CognomeS
having count(*) ≥ 5
```
- (b)

```
select A.NomeA, A.Livello
from ATTIVITA A
where IdA IN (select L.IdA
              from LEZIONI L, SOCI S, ISCRIZIONI-LEZIONI IL
              where L.Data ≥ 1/03/04 and L.Data ≤ 1/05/04 and
                    IL.NumTessera=S.NumTessera and IL.IdL=L.IdL
                    and IL.DataIscr ≥ 1/01/04 and IL.DataIscr ≤ 20/01/04
                    and IL.Acconto ≤ 50 and S.ScadenzaT ≤ 2/05/04
              group by IL.NumTessera,L.IdA
              having sum(L.Costo) ≥ 500)
```

Separatamente per ciascuna delle due interrogazioni SQL

- (a) Si scriva l'espressione algebrica corrispondente.
- (b) Si scriva il piano di esecuzione che potrebbe essere scelto dall'ottimizzatore in assenza di strutture fisiche per migliorare le prestazioni dell'interrogazione.
- (c) Si scelgano una o più strutture fisiche accessorie per migliorare le prestazioni dell'interrogazione. Si motivi la scelta e si definisca il nuovo piano di esecuzione ipotizzato.

NOTA BENE: Sia per il punto b) che per il punto c)

- dove necessario, si ipotizzi la distribuzione dei dati.
 - nel descrivere il piano di esecuzione si indichi quali operazioni sono svolte, in che ordine e, dove possibile, quale tipo di operatore è usato per svolgere una certa operazione (ad esempio in caso di join).
2. (13 Punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con *):

```

SERRA(CodSerra, Locazione, NumSensori)
SENSORE(CodSensore, GrandezzaMisurata, CodSerra, OffsetCritico)
LOG-EVENTI(CodE, TimeStamp, TipoEvento, CodSensore, Valore)
MISURA(CodSensore, TimeStamp, Valore)
SINTESI-GIORNO(Data, CodSensore, MediaValore)
NOTIFICA(CodN, CodSerra, Locazione, Messaggio)

```

Si scriva il trigger che gestisce l'arrivo di un nuovo evento di tipo misura (`TipoEvento='M'`). Per questo tipo di eventi, occorre inserire una nuova misura nella tabella `MISURA`. Inoltre, occorre verificare se si è verificata una situazione critica nella serra dove si trova il sensore che ha effettuato la misura.

Una situazione è critica se più di metà dei sensori presenti nella serra considerata hanno un valore eccessivo per l'ultima misura effettuata in ordine di tempo. Il valore di una misura è eccessivo se è maggiore della misura media del giorno per quel sensore di una quantità superiore a `OffsetCritico`. La misura media del giorno è memorizzata nella tabella `SINTESI-GIORNO` per ogni sensore. Questa tabella è mantenuta automaticamente aggiornata da un trigger non considerato nella presente applicazione. Per estrarre la data dall'attributo `TimeStamp`, si utilizzi una generica funzione `DATE(TimeStamp)`.

Se si verifica una situazione critica, occorre inserire una richiesta di notifica nella tabella `NOTIFICA`. Il codice identificativo univoco `CodN` è un contatore che deve essere incrementato ogni volta che è inserita una nuova notifica.

3. (4 Punti) Si consideri il seguente file di log relativo ad un nodo nel caso di una base di dati distribuita. Applicare il protocollo di ripresa a caldo dopo la caduta del nodo assumendo un algoritmo di commit a due fasi, a fronte del seguente input (dove $r(t_i)$ indica la presenza di un record ready per la transazione t_i)
- ```

b(t1) i(t1 o1 a1) b(t2) u(t2 o2 b2 a2) r(t1) b(t3) c(t1) u(t3 o3 b3 a3)
r(t2) b(t4) CKPT(t2,t3,t4) c(t2) b(t5) d(t3 o6 b6) r(t3) a(t3) i(t4 o4 a4)
r(t4) u(t5 o5 b5 a5) c(t4) r(t5) guasto

```

# Esame di Tecnologia delle basi di dati

6 Maggio 2006

1. (13 punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate):

```
FORNITORI(CodF, NomeF, Indirizzo, Città)
PRODOTTI(CodP, NomeP, Prezzo, Categoria)
DISTRIBUTORI(CodD, Modello, DataCostruzione, Luogo, MaxNumProdotti)
FORNITURE_DISTRIBUTORI(CodF, CodP, CodD, Data, Quantità)
```

Si ipotizzino le seguenti cardinalità per le tabelle:

- $\text{card}(\text{FORNITORI}) \simeq 10^4$  tuple,  
 $\text{card}(\sigma_{\text{Città}='Torino'} \text{FORNITORI}) \simeq 10^2$  tuple,
- $\text{card}(\text{PRODOTTI}) \simeq 10^4$  tuple,  
 $\text{card}(\sigma_{\text{Categoria}='Sigarette'} \text{PRODOTTI}) \simeq 10^2$  tuple,  
 $\text{card}(\sigma_{\text{Prezzo}=2} \text{PRODOTTI}) \simeq 10^3$  tuple,
- $\text{card}(\text{DISTRIBUTORI}) \simeq 10^3$  tuple,  
 $\text{card}(\sigma_{\text{MaxNumProdotti} \leq 100} \text{DISTRIBUTORI}) \simeq 10$  tuple,
- $\text{card}(\text{FORNITURE\_DISTRIBUTORI}) \simeq 10^8$  tuple per l'anno 2005,  
 $\text{card}(\sigma_{\text{Quantità} > 100} \text{FORNITURE\_DISTRIBUTORI}) \simeq 10^6$  tuple per l'anno 2005.

Si considerino le seguenti query SQL:

- (a) 

```
select NomeF, Indirizzo
from FORNITORI F
where Città='Torino' and CodF IN
 (Select CodF
 from DISTRIBUTORI D, PRODOTTI P, FORNITURE_DISTRIBUTORI FD
 where P.CodP=FD.CodP and D.CodD=FD.CodD and P.Categoria='Sigarette'
 and Prezzo ≥ 2.5 and (D.Modello='XY-123' or D.Modello='YZ-345')
 and Data ≥ 1/11/05 and Data ≤ 1/12/05
 GROUP BY CodF
 HAVING SUM(Quantità) > 5000)
```
- (b) 

```
select P.Nome, P.Categoria
from FORNITURE-DISTRIBUTORI FD, PRODOTTI P, FORNITORI F
where FD.CodP=P.CodP and FD.CodF=F.CodF and
and P.Categoria <> 'snack' and FD.Data ≥ 1/02/05 and FD.Data ≤ 1/08/05
and FD.Quantità > 100 and P.Prezzo=2 and Città <> 'Milano'
and not exists (select *
 from DISTRIBUTORI D
 where D.CodD=FD.CodD and MaxNumProdotti ≤ 100)
group by P.CodP, FD.CodD, P.NomeP, P.Categoria
having COUNT(*) ≥ 20
```

Separatamente per ciascuna delle due interrogazioni SQL

- (a) Si scriva l'espressione algebrica corrispondente.
- (b) Si scriva il piano di esecuzione che potrebbe essere scelto dall'ottimizzatore in assenza di strutture fisiche per migliorare le prestazioni dell'interrogazione.

- (c) Si scelgano una o più strutture fisiche accessorie per migliorare le prestazioni dell'interrogazione. Si motivi la scelta e si definisca il nuovo piano di esecuzione ipotizzato.

NOTA BENE: Sia per il punto b) che per il punto c)

- dove necessario, si ipotizzi la distribuzione dei dati.
- nel descrivere il piano di esecuzione si indichi quali operazioni sono svolte, in che ordine e, dove possibile, quale tipo di operatore è usato per svolgere una certa operazione (ad esempio in caso di join).

2. (13 Punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con \*):

```
CLIENTE(CodC, NomeC, NumTel)
MALFUNZIONAMENTO(CodM, CodS, CodC, TimeStamp)
CHIAMATA(IdCH, NumChiamante, TipoChiamata, CodS, TimeStamp)
CONTRATTO(CodCT, CodC, DataInizio, DataFine, CodS, CodG)
SERVIZIO(CodS, Descr, Costo)
GESTORE(CodG, NomeG, Sede)
CRITICITÀ(CodS, CodG, Data, NumSegnalazioni)
```

Si scriva il trigger che gestisce l'arrivo delle chiamate in un call center. Le chiamate possono essere di varie tipologie (richieste di informazioni, segnalazioni di malfunzionamenti di servizi, ...). Occorre gestire le chiamate che segnalano malfunzionamenti di servizi erogati dall'azienda ( $TipoChiamata = 'M'$ ).

Prima di tutto, occorre verificare che la segnalazione sia corretta. In particolare, la segnalazione deve essere stata fatta da un cliente dell'azienda che, per la data in cui avviene la segnalazione, ha un contratto per il servizio di cui segnala il malfunzionamento. Per estrarre la data dall'attributo `TimeStamp`, si utilizzi una generica funzione `DATE(TimeStamp)`. Se la segnalazione è corretta, occorre inserirla nella tabella `MALFUNZIONAMENTO`. Il codice identificativo univoco `CodM` è un contatore che deve essere incrementato ogni volta che è inserito un nuovo malfunzionamento.

Il servizio può essere erogato a clienti diversi da gestori diversi (si veda la tabella `CONTRATTO`). In un dato periodo di tempo (la durata del contratto), un cliente può fruire del servizio da parte di un solo gestore. Per verificare la qualità del servizio erogato, si vuole confrontare il numero di segnalazioni di malfunzionamenti ricevuti durante la giornata per quel servizio e quel gestore con la media per quel servizio durante la stessa giornata (calcolata come rapporto tra il numero totale di segnalazioni ricevute e il numero di gestori a cui facevano riferimento). Se il numero di segnalazioni del gestore è superiore alla media, le informazioni sullo stato di erogazione del servizio per quel giorno (gestore, servizio, data, numero totale di segnalazioni ricevute quel giorno) sono inserite in una tabella di segnalazioni di criticità (`CRITICITÀ`). Se la terna (gestore, servizio, data) è già presente (cioè è già avvenuto l'inserimento in precedenza durante la stessa giornata), occorre invece incrementare il numero di segnalazioni indicato.

3. (4 Punti)

Si consideri il seguente file di log relativo ad un nodo nel caso di una base di dati distribuita. Applicare il protocollo di ripresa a caldo dopo la caduta del nodo assumendo un algoritmo di commit a due fasi, a fronte del seguente input (dove  $r(t_i)$  indica la presenza di un record ready per la transazione  $t_i$ )

b(t1) b(t2) b(t5) i(t5 o5 a2) b(t8) i(t2 o2 a3) r(t5) u(t1 o1 b1 a1) c(t5) i(t8 o3 a3)  
CKPT(t1,t2,t8) i(t1 o6 a6) u(t8 o8 b5 a5) r(t2) r(t1) b(t4) a(t1) i(t4 o9 a6) c(t2) b(t3)  
d(t8 o7 b3) i(t3 o4 a1) r(t8) d(t4 o10 b3) r(t3) c(t8) r(t4) guasto

# Esame di Tecnologia delle basi di dati

20 Aprile 2007

1. (13 punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate):

```
SEDE(CodS, Nazione, Città, Indirizzo)
LABORATORIO(CodL, CodS, NomeLab, Superficie, NumDipendenti)
RICERCATORE(CodR, CodL, NomeR, DataNascita, Ruolo)
ATTIVITÀ(CodR, CodP, NomeA, DataInizio, DataFine, Stato, DescrizioneA, Costo)
PROGETTO(CodP, NomeP, Budget, Durata, Acronimo, Logo, DescrizioneP)
```

Siano note le seguenti cardinalità e distribuzioni per le tabelle:

- $\text{card}(\text{SEDE}) \simeq 10^2$  tuple,  $\text{card}(\sigma_{\text{Nazione}='Italia'} \text{SEDE}) \leq 10$  tuple
- $\text{card}(\text{LABORATORIO}) \simeq 10^4$  tuple,  
 $\text{MIN}(\text{NumDipendenti})=1$ ,  $\text{AVG}(\text{NumDipendenti})=10$ ,  $\text{MAX}(\text{NumDipendenti})=100$ ,  
 $\text{MIN}(\text{Superficie})=100$ ,  $\text{AVG}(\text{Superficie})=500$ ,  $\text{MAX}(\text{Superficie})=1000$
- $\text{card}(\text{RICERCATORE}) \simeq 10^6$  tuple,  
 $\text{MIN}(\text{DataNascita})=1949$ ,  $\text{MAX}(\text{DataNascita})=1984$ ,  
 $\text{card}(\sigma_{\text{Ruolo}='Responsabile'} \text{RICERCATORE}) \simeq 4 \cdot 10^4$  tuple,  
 $\text{card}(\sigma_{\text{Ruolo}='ViceResponsabile'} \text{RICERCATORE}) \simeq 5 \cdot 10^4$  tuple
- $\text{card}(\text{ATTIVITÀ}) \simeq 5 \cdot 10^7$  tuple,  
 $\text{MIN}(\text{DataInizio})=1994$ ,  $\text{AVG}(\text{Costo})=10000$ ,  
 $\text{card}(\sigma_{\text{Stato}='Terminato'} \text{ATTIVITÀ}) \simeq 4 \cdot 10^7$  tuple
- $\text{card}(\text{PROGETTO}) \simeq 10^5$  tuple,  
 $\text{MIN}(\text{Durata})=24$ ,  $\text{AVG}(\text{Durata})=30$ ,  $\text{MAX}(\text{Durata})=48$

Si considerino le seguenti query SQL:

- (a) 

```
SELECT R.CodR, P.CodP, NomeR, NomeP
FROM ATTIVITÀ A, RICERCATORE R, PROGETTO P
WHERE R.CodR=A.CodR AND P.CodP=A.CodP
AND Stato<>'Terminato' AND Budget<90000 AND Durata>47
AND (Ruolo='Responsabile' OR DataNascita<1980) AND EXISTS
 (SELECT * FROM LABORATORIO L
 WHERE L.CodL=R.CodL
 AND NumDipendenti<>10 AND Superficie>985)
ORDER BY P.CodP, NomeP
```
- (b) 

```
SELECT P.CodP, NomeP
FROM ATTIVITÀ A, PROGETTO P
WHERE P.CodP=A.CodP
AND (DataInizio≥2007 OR DataInizio≤1994) AND A.CodR IN
 (SELECT CodR FROM RICERCATORE R, LABORATORIO L, SEDE S
 WHERE L.CodL=R.CodL AND S.CodS=L.CodS
 AND NumDipendenti<20 AND Nazione='Italia'
 AND DataNascita<1980 AND DataNascita>1965
 AND (Ruolo='Responsabile' OR Ruolo='ViceResponsabile'))
GROUP BY P.CodP, NomeP
HAVING SUM(Costo)<8000
```



- (b) si scriva il piano di esecuzione che potrebbe essere scelto dall'ottimizzatore in assenza di strutture fisiche per migliorare le prestazioni dell'interrogazione;
- (c) si scelgano una o più strutture fisiche accessorie per migliorare le prestazioni dell'interrogazione. Si motivi la scelta e si definisca il nuovo piano di esecuzione ipotizzato.

NOTA BENE: per i punti b) e c), dove necessario, si ipotizzi la distribuzione dei dati; nel descrivere il piano di esecuzione si indichi quali operazioni sono svolte, in che ordine e, dove possibile, quale tipo di operatore è usato per svolgere una certa operazione (ad esempio in caso di join).

2. (13 Punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate, gli attributi opzionali hanno l'asterisco):

RICHIESTA\_BOLLETTA(CodR, CodC, DataRichiestaBolletta, DataInizioBolletta, DataFineBolletta)  
 CLIENTI-SOCIETÀ(CodC, Nome, Indirizzo, Città, NumTel)  
 TRAFFICO-TELEFONICO(NumTelChiamante, NumTelChiamato, TimeStamp, FasciaOraria, DurataInMinuti)  
 TARIFFE(FasciaOraria, TipoOperatore, CostoAlMinuto)  
 BOLLETTA(CodB, CodC, DataEmissione, ImportoTotale, NumeroTotaleUtentiDiversiChiamati, NumeroTotaleChiamateEffettuate)  
 ATTIVAZIONE-PROMOZIONE(CodA, NomePromozione, TipoPremio)  
 PREMI-DISPONIBILI(CodP, TipoPremio, Valore)  
 PREMI-ASSEGNATI(CodA, CodP, CodC)

Si scrivano i trigger necessari per gestire alcune attività di una società telefonica.

La prima attività è la gestione della richiesta di emissione della bolletta telefonica per un certo cliente e per il traffico telefonico relativo ad uno specifico intervallo di tempo. L'informazione sul traffico telefonico effettuato dai clienti della società è disponibile nella tabella TRAFFICO-TELEFONICO. Per estrarre la data dall'attributo TimeStamp si utilizzi una generica funzione DATE(TimeStamp).

La tabella TARIFFE contiene le tariffe applicate. La tariffa varia a seconda della fascia oraria in cui si effettua la chiamata e dal tipo di operatore utilizzato. Se il numero chiamato fa parte dei clienti della società si applica la tariffa corrispondente a TipoOperatore='DellaSocietà'. Se invece il numero chiamato non fa parte dei clienti della società si applica la tariffa corrispondente a TipoOperatore='AltroOperatore'. Ciascuna bolletta è identificata da un codice univoco progressivo (CodB) per ogni cliente. Quando una bolletta viene emessa, si deve eliminare tutto il traffico telefonico considerato per emettere la bolletta.

La seconda attività è la gestione delle promozioni periodicamente attivate dalla società allo scopo di incentivare la fidelizzazione dei clienti. Si vogliono premiare i clienti per cui si è registrato un aumento del traffico telefonico. In particolare si vuole premiare il cliente per cui l'importo dell'ultima bolletta emessa è superiore all'importo di ciascuna delle bollette precedenti. (si assuma che ci sia al più un cliente che soddisfa questi requisiti). Al cliente viene assegnato il prodotto di valore massimo tra i prodotti disponibili del tipo indicato nella promozione (si assuma che ci sia un solo prodotto che soddisfa i requisiti).

3. (4 Punti) Stabilire l'ordine di esecuzione mediante lock a due fasi per il seguente schedule, evidenziando anche lo stato di locking dei dati durante l'esecuzione:

r3(C) r2(M) w4(M) r4(A) r1(B) r1(A) w2(B) w3(C) r1(C) r3(A)

Si ipotizzi che dopo l'ultima operazione nello schedule ciascuna transazione esegua un commit.

# Esame di Tecnologia delle Basi di Dati

24 Aprile 2008

1. (13 punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate):

```
CATEGORIA(CCodice, CNome, Descrizione)
FARMACO(FCodice, Nome, PrincipioAttivo, CCodice, Produttore)
FARMACIA(FARCodice, Proprietario, Indirizzo, Città)
FORNITURE(FARCodice, FCodice, Data, Qta)
```

Si ipotizzino le seguenti cardinalità per le tabelle:

- $\text{card}(\text{CATEGORIA}) \simeq 10^2$  tuple,
- $\text{card}(\text{FARMACO}) \simeq 10^5$  tuple,  
 $\text{card}(\sigma_{\text{PrincipioAttivo}='Paracetamolo'} \text{ FARMACO}) \simeq 10^3$  tuple,  
numero di produttori di farmaci  $\simeq 20$ ,
- $\text{card}(\text{FARMACIA}) \simeq 10^4$  tuple,  
 $\text{card}(\sigma_{\text{Città}='Roma'} \text{ FARMACIA}) \simeq 10^2$  tuple,
- $\text{card}(\text{FORNITURE}) \simeq 10^8$  tuple per l'anno 2007,

Si considerino le seguenti query SQL:

- (a) 

```
select FAR.FarCodice, F.FCodice
from FARMACIA FAR, FORNITURE FT, FARMACO F
where FAR.FarCodice = FT.FarCodice and FT.FCodice=F.FCodice
and FT.Data ≥ 01-01-07 and FT.Data ≤ 31-01-07
and PrincipioAttivo='Paracetamolo'
group by FAR.FARCodice, F.FCodice
having sum(Qta) ≥ 300
```
- (b) 

```
select CNome
from CATEGORIA C, FARMACO F
where C.CCodice=F.CCodice and F.FCodice in (select FT.FCodice
from FARMACIA FAR, FORNITURE FT, FARMACO F
where FAR.FARCodice = FT.FARCodice and F.FCodice=FT.FCodice
and FT.Data ≥ 01-03-07 and FT.Data ≤ 31-10-07
and Città='Roma' and Produttore='Bayer'
group by FT.FCodice
having count(distinct FAR.FARCodice) > 10)
```

Separatamente per ciascuna delle due interrogazioni SQL

- Si scriva l'espressione algebrica corrispondente, indicando le operazioni svolte, la cardinalità e la selettività di ogni operazione. Dove necessario, si ipotizzi la distribuzione dei dati.
- Si scriva il piano di esecuzione che potrebbe essere scelto dall'ottimizzatore in assenza di strutture fisiche accessorie (ordine e tipo dei join, accesso alle tabelle).
- Si scelgano le strutture fisiche accessorie per migliorare le prestazioni dell'interrogazione. Si motivi la scelta e si definisca il nuovo piano di esecuzione ipotizzato.

2. (13 Punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate, gli attributi opzionali hanno l'asterisco):

```
RICHIESTA_ACQUISTO_LIBRI(CodRichiestaAcquisto, CodCliente, Data, ISBN, NumCopieRichieste)
CATALOGO_LIBRI(ISBN, Titolo, Autore, Prezzo)
MAGAZZINO_LIBRI(ISBN, NumCopieDisponibili)
NOTIFICA_ACQUISTO(CodN, Data, Messaggio, Titolo, Autore, Prezzo)
RICHIESTA_INFORMAZIONI_AUTORE(CodRichiestaInformazioni, CodCliente, Data, Autore)
VENDITE_LIBRI(ISBN, NumCopieVendute)
NOTIFICA_INFORMAZIONI(CodRichiestaInformazioni, CodCliente, ISBN, Titolo, Autore)
```

Si scrivano i trigger necessari per gestire le seguenti due attività in un sito per la vendita on-line di libri: (1) richiesta di acquisto di libri e (2) richiesta di informazioni su un autore. Il sito dispone di un catalogo dei libri venduti presso il sito (tabella CATALOGO\_LIBRI).

La prima attività è la gestione della richiesta di acquisto di un libro a catalogo da parte di un cliente (inserimento nella tabella RICHIESTA\_ACQUISTO\_LIBRI). Nella richiesta di acquisto il cliente specifica il libro e il numero di copie che desidera acquistare. Se tutte le copie richieste sono disponibili nel magazzino, la richiesta di acquisto può essere completata. In questo caso viene aggiornata la disponibilità nel magazzino per il libro, e si notifica al cliente l'avvenuto acquisto (Messaggio da notificare: "Acquisto concluso"). Altrimenti si notifica al cliente il mancato acquisto (Messaggio da notificare: "Acquisto non concluso"). Si consideri che la tabella MAGAZZINO\_LIBRI contiene, tra i libri disponibili nel catalogo, solo quelli per cui è disponibile almeno una copia nel magazzino. I libri per cui non ci sono copie disponibili non sono presenti in MAGAZZINO\_LIBRI. L'attributo CodN nella tabella NOTIFICA\_ACQUISTO è un contatore che deve essere incrementato ogni volta che viene effettuata una nuova notifica in uno stesso giorno.

La seconda attività consiste nel fornire informazioni ai clienti sui libri venduti presso il sito, per supportare i clienti nella scelta del libro da acquistare. Quando un cliente richiede informazioni su un autore (inserimento nella tabella RICHIESTA\_INFORMAZIONI\_AUTORE), viene selezionato il libro che ha venduto più copie, tra i libri scritti dall'autore e per cui è disponibile almeno una copia nel magazzino. Si assuma che ci sia sempre al più un unico libro che soddisfi tutti i requisiti. Le informazioni sul libro selezionato sono notificate al cliente (inserimento nella tabella NOTIFICA\_INFORMAZIONI).

3. (4 Punti) Si consideri il seguente file di log relativo ad un nodo nel caso di una base di dati distribuita. Applicare il protocollo di ripresa a caldo dopo la caduta del nodo assumendo un algoritmo di commit a due fasi, a fronte del seguente input (dove r(ti) indica un record ready per la transazione ti)
- ```
b(t6) d(t6 o6 b8) i(t6 o1 a1) b(t1) d(t1 o5 b2) b(t5) u(t6 o7 b9 a9) u(t5 o8 b1 a1) r(t1)
a(t1) CKPT(t6,t5) b(t2) i(t2 o10 a9) r(t6) b(t8) u(t8 o9 b6 a6) a(t6) d(t5 o5 b4) r(t2)
b(t7) c(t2) d(t7 o4 b1) i(t5 o3 a4) r(t5) r(t7) c(t5) r(t8) guasto
```

Esame di Tecnologia delle basi di dati

1 Luglio 2008

1. (13 punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate):

```
STAZIONE(CodS, NomeS, Città)
BIGLIETTO(CodB, CodV, CodSPartenza, CodSArrivo, Costo, DataEmissione)
VIAGGIO(CodV, DataV, CodT, CodSPartenza, CodSArrivo, OraPartenza, OraArrivo)
TRENO(CodT, TipoTreno, #Posti, #Carrozze)
```

Si ipotizzino le seguenti cardinalità per le tabelle:

- $\text{card}(\text{STAZIONE}) \simeq 10^3$ tuple,
- $\text{card}(\text{BIGLIETTO}) \simeq 10^8$ tuple per l'anno 2007,
 $\text{card}(\sigma_{\text{Costo} \leq 30} \text{BIGLIETTO}) \simeq 10^6$ tuple,
- $\text{card}(\text{VIAGGIO}) \simeq 10^6$ tuple per l'anno 2007,
 $\text{card}(\text{VIAGGIO}) \simeq 5 \cdot 10^3$ tuple per la stazione di arrivo Porta Susa,
 $\text{card}(\text{VIAGGIO}) \simeq 10^4$ tuple per la stazione di partenza Milano Centrale,
- $\text{card}(\text{TRENO}) \simeq 10^3$, i tipi di treno sono 15.

Si considerino le seguenti query SQL:

- (a)

```
select CodV, SUM(costo)
from BIGLIETTO
where DataEmissione ≥ 01-07-07 and DataEmissione ≤ 31-08-07
and CodV not in (select CodV
                  from VIAGGIO V, STAZIONE S, TRENO T
                  where V.CodSPartenza=S.CodS and V.CodT=T.CodT
                  and TipoTreno='Regionale'
                  and NomeS='Milano Centrale')
```

group by CodV
- (b)

```
select DataV, COUNT(*)
from VIAGGIO V, BIGLIETTO B, STAZIONE S, TRENO T
where V.CodSArrivo=S.CodS and B.CodV=V.CodV
and DataV ≥ 01-01-07 and DataV ≤ 31-10-07
and Costo ≤ 30 and T.CodT=V.CodT
and TipoTreno <> 'Eurostar' and NomeS='Porta Susa'
group by DataV
```

Separatamente per ciascuna delle due interrogazioni SQL

- Si scriva l'espressione algebrica corrispondente.
- Si scriva il piano di esecuzione che potrebbe essere scelto dall'ottimizzatore in assenza di strutture fisiche accessorie.
- Si scelgano una o più strutture fisiche accessorie per migliorare le prestazioni dell'interrogazione. Si motivi la scelta e si definisca il nuovo piano di esecuzione ipotizzato.

NOTA BENE: Sia per il punto b) che per il punto c)

- dove necessario, si ipotizzi la distribuzione dei dati,

- nel descrivere il piano di esecuzione si indichino le cardinalità, quali operazioni sono svolte, in che ordine e, dove possibile, quale tipo di operatore è usato per svolgere l'operazione (es. tipo di join).
2. (13 Punti) Sono date le relazioni seguenti (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con *):

```

CELLA(CodCella, x0, y0, x1, y1, NumTelPresenti, MaxChiamate)
TELEFONO(NumTel, x, y, StatoTelefono)
VARIAZIONE-STATO(CodV, TimeStamp, NumTel, x, y, TipoVariazione)
LOG-ECCEZIONI(CodE, CodCella, TipoEccezione)

```

Lo schema descrive una rete di telefonia cellulare. I telefoni all'interno della rete sono descritti nella tabella **TELEFONO**. Lo stato di un telefono all'interno della rete può assumere i valori **acceso** o **attivo** (il telefono è **attivo** se sta eseguendo una chiamata). Quando il telefono è spento non è presente nella tabella **TELEFONO**. La posizione corrente all'interno della rete di ogni telefono è definita dalle sue coordinate x, y .

La rete di telefonia cellulare è composta da celle. Per semplicità si considerino celle quadrate. Le celle della rete sono descritte nella tabella **CELLA**. La posizione di ogni cella all'interno della rete è definita dalle coordinate x_0, y_0 del suo estremo inferiore sinistro e dalle coordinate x_1, y_1 del suo estremo superiore destro. La cella di appartenenza di un telefono è l'unica cella che soddisfa entrambe le condizioni $x_0 \leq x < x_1$ e $y_0 \leq y < y_1$. Per ogni cella è noto il numero di telefoni correntemente presenti nella cella (attributo **NumTelPresenti**), e il massimo numero di telefoni che possono effettuare una chiamata (attributo **MaxChiamate**).

Si scrivano i trigger che gestiscono le seguenti variazioni di stato della rete (inserimenti nella tabella **VARIAZIONE-STATO**).

Accensione/spengimento di un telefono. I tipi di variazione sono 'A' (acceso) e 'S' (spento). Quando il telefono viene acceso, occorre introdurre l'informazione su di esso nella tabella **TELEFONO**, mentre quando viene spento occorre eliminarla. Inoltre, occorre localizzare la cella di appartenenza del telefono e modificare il conteggio dei telefoni presenti (attributo **NumTelPresenti**).

Spostamento di posizione di un telefono. Il tipo di variazione è 'L' mentre x e y rappresentano le coordinate della nuova posizione del telefono. Occorre aggiornare la posizione del telefono. Inoltre, occorre verificare se lo spostamento di posizione del telefono richiede un cambiamento di cella. In questo caso, devono essere modificati i contatori dei telefoni presenti delle due celle interessate.

Inizio di una telefonata. Il tipo di variazione è 'I'. Se la cella in cui è localizzato il telefono non supera il numero massimo di chiamate che può gestire (attributo **MaxChiamate**), occorre modificare lo stato del telefono, che diviene 'attivo'. Se invece la cella supera il numero di chiamate che è in grado di gestire, il telefono non può essere attivato. Occorre inserire l'informazione che il numero di chiamate è stato superato nella tabella **LOG-ECCEZIONI**. Il codice **CodE** è un contatore, univoco all'interno di ogni cella.

3. (4 Punti) Si consideri il seguente schedule:
- ```

r4(z) w4(z) r0(z) w0(z) w2(y) w0(x) w4(y) r2(x) w2(y)

```

Indicare se lo schedule è VSR o CSR serializzabile. Se serializzabile, indicare un'esecuzione seriale corrispondente.