

Basi di Dati e Sistemi Informativi I, Prove scritte AA 2003/04

Adriano Peron

Facoltà di Scienze M.F.N., Corso di Laurea in Informatica, Dipartimento di Scienze Fisiche,
Università di Napoli 'Federico II', Italy
E-mail: peron@na.infn.it

1 Prova scritta del 8-09-04

Si consideri il seguente schema relazionale per la descrizione bibliografica di articoli pubblicati su riviste scientifiche:

$RIVISTA(COD - R, TITOLO - R, EDITORE)$

$ARTICOLO(COD - AR, COD - R, TITOLO, NUMERO, ANNO, PAG - I, PAG - F)$

$AUTORI(COD - AU, Nome, Cognome)$

$SCRIVE(COD - AR, COD - AU)$

$CITAZIONI(COD - CITATO, COD - RIF)$.

Una istanza di $ARTICOLO$ ha come attributi il codice dell'articolo $COD - AR$ (chiave) il codice della rivista, il titolo dell'articolo, il numero del fascicolo della rivista, l'anno, la pagina iniziale e finale; Un articolo può avere uno o più autori. Gli autori sono descritti dallo schema $AUTORI$ e lo schema $SCRIVE$ codifica il legame molti a molti tra autori e articoli: se xx è il codice di un autore e yy è il codice di un articolo, una istanza (yy, xx) di $SCRIVE$ indica che xx è un autore di yy . $CITAZIONI$ memorizza i riferimenti tra articoli: se (zz, yy) è una istanza di $CITAZIONI$, allora yy è il codice di un articolo che cita un altro articolo zz nella sua bibliografia.

Esercizio 11 Si scriva (senza usare operazioni di conteggio) una espressione in algebra relazionale che, se valutata, fornisce nome, cognome degli autori che non abbiano mai scritto da soli un articolo (gli articoli da loro scritti hanno sempre più di un autore. Suggerimento: si trovino prima gli articoli scritti da un solo autore ...).

Esercizio 12 Scrivere una interrogazione SQL che fornisca titolo di rivista e titolo dell'articolo pubblicato nella rivista avente massimo numero di citazioni.

Esercizio 13 Si scriva una procedura DescrizioneBibliografica in C o Pascal che riceve in ingresso il codice di un articolo e restituisce in uscita una stringa di testo che rappresenta la descrizione bibliografica completa dell'articolo; la stringa fornita in uscita deve avere la seguente struttura: <lista cognome nome degli autori in ordine alfabetico>, <titolo articolo>, <titolo rivista>, <numero>, <anno>, <pagina iniziale>, <pagina finale>.

Sfruttando la procedura DescrizioneBibliografica si scriva una procedura Bibliografia che riceve in ingresso il codice di un articolo e restituisce in uscita una stringa di testo ottenuta concatenando la descrizione bibliografica completa di tutti gli articoli citati nell'articolo passato come parametro.

- Esercizio 14**
1. Si dica che cosa si intende con indice primario;
 2. Un indice primario è denso (giustificare la risposta)?
 3. Si descrivano la struttura ed i vantaggi di un indice multilivello.

Soluzione Esercizio 11

Siano RI , ART , AU , SCR CIT le relazioni per RIVISTE, ARTICOLI, AUTORI, SCRIVE e CITAZIONE rispettivamente.

Articoli scritti da più di un autore.

$$PIU \leftarrow \Pi_{COD-AR}(\rho_{SCR1(COD-AR1, COD-AU1)}(SCR) \bowtie_C SCR)$$

dove C è la condizione $COD - AR = COD - AR1 \wedge COD - AU <> COD - AU1$.

Articoli scritti da un solo autore.

$$SOLO \leftarrow \Pi_{COD-AR}(ART) \setminus PIU$$

Autori che hanno scritto un articolo da soli

$$AUTORESULO \leftarrow \Pi_{COD-AU}(SCRIVE \bowtie SOLO)$$

Soluzione

$$\Pi_{Nome,Cognome}(AU \bowtie (\Pi_{COD-AU}(AU) \setminus AUTORESULO))$$

† Soluzione Esercizio 12

Creazione di una vista per il conteggio del numero di citazioni per articolo (con almeno una citazione).

```
CREATE VIEW NCitazioni(COD-AR,NCit) AS
SELECT COD-CITATO, COUNT(*)
FROM CITAZIONI
GROUP BY COD-CITATO
```

Se si vogliono far comparire anche gli articoli senza citazione (NCit=0) la vista può essere definita nel seguente modo

```
CREATE VIEW NCitazioni(COD-AR,NCit) AS
SELECT COD-AR, COUNT(COD-RIF)
FROM ARTICOLI RIGHT OUTER JOIN CITAZIONI ON COD-ART=COD-CITATO
GROUP BY COD-AR
```

Soluzione

```
SELECT R.TITOLO, A.TITOLO
FROM RIVISTA AS R NATURAL JOIN ARTICOLO AS A NATURAL JOIN NCitazioni
WHERE NCit >= ALL (SELECT NCit
                    FROM ARTICOLO AS A1 NATURAL JOIN NCitazioni
                    WHERE A1.COD-R = A.COD-R)
```

†

Soluzione Esercizio 13

per semplicità si supponga che tutti campi della descrizione dell'articolo siano di tipo stringa.

```
PROCEDURE BibItem(IN Art : string, OUT bib : string)
```

```
VAR
EXEC SQL BEGIN DECLARE SECTION
        titA, titR, NomeA, CognomeA, AnnoA, PagI, PagF : string;
        Num, Cart : string
EXEC SQL END DECLARE SECTION
```

```

BEGIN
Cart := Art; bib:= "";
EXEC SQL DECLARE scan CURSOR FOR
    SELECT NOME, COGNOME
    FROM   SCRIVE NATURAL JOIN AUTORI
    WHERE  COD-AR = :Cart
    ORDER BY COGNOME, NOME;
EXEC SQL OPEN scan;
EXEC SQL FETCH scan INTO :NomeA, :CognomeA
WHILE SQLCODE = 0 DO
    BEGIN
    bib := bib + :NomeA + ', ' + :CognomeA + ', ';
    EXEC SQL FETCH scan INTO :NomeA, :CognomeA;
    END
EXEC SQL SELECT A.TITOLO, R.TITOLO, A.NUMERO, A.ANNO, A.PAG-I, A.PAG-F
    INTO :titA, :titR, :Num, :AnnoA, :PagI, :PagF
    FROM   ARTICOLO AS A NATURAL JOIN RIVISTA AS R
    WHERE  COD-AR = :Cart
IF SQLCODE = 0 THEN bib := bib + titA + ', ' + titR + ', '
    + :Num + ', ' + :AnnoA + ', ' + :PagI + ', ' + :PagF + ', '
END

PROCEDURE Bibliografia(IN Art : string, OUT bib : string)

VAR
EXEC SQL BEGIN DECLARE SECTION
    Aus, Cart, Citato : string
EXEC SQL END DECLARE SECTION

BEGIN
Cart := Art; bib:= "";
EXEC SQL DECLARE scan CURSOR FOR
    SELECT COD-CITATO
    FROM   CITAZIONI
    WHERE  COD-RIF = :Cart
EXEC SQL OPEN scan;
EXEC SQL FETCH scan INTO :Citato
WHILE SQLCODE = 0 DO
    BEGIN
    Bibitem(Citato,Aus)
    bib := bib + Aus + ' ';
    EXEC SQL FETCH scan INTO :Citato;
    END
END

```

2 Prova scritta del 28-07-04

Si consideri il seguente schema relazionale che descrive la gestione di sale cinematografiche:

SALA(CodSala, Nome, Schermo, Audio)

POSTO(CodSala, CodPosto, Fila, NumPosto, Fascia)

PROIEZIONE(CodicePr, Film, CodSala, Data, OraInizio)

BIGLIETTO(CodicePr, CodPosto).

SALA, fornisce alcune caratteristiche tecniche della sala di proiezione;

POSTO descrive i singoli posti a sedere di una sala qualificandoli con il codice della sala (CodSala), il numero della fila (Fila), il numero del posto nella fila (NumPosto), il livello di qualità del posto (Fascia che può assumere i valori 'A', 'B' o 'C', dove 'A' indica la fascia di livello superiore);

PROIEZIONE descrive le proiezioni dei film nelle sale qualificandoli con il film proiettato, il codice della sala, la data e l'ora della proiezione;

BIGLIETTO contiene informazione sui posti venduti (CodPosto) per ogni proiezione (CodicePr).

Esercizio 21 *Si scriva una espressione in algebra relazionale che, se valutata, fornisce il codice delle proiezioni per le quali tutti i posti di fascia 'A' sono stati occupati.*

Esercizio 22 *Scrivere una vista in SQL che per ogni proiezione fornisca la data, il codice della sala, l'ora di proiezione, il numero di posti occupati e il numero di posti liberi (separatamente) per fascia 'A', 'B' e 'C'.*

Esercizio 23 *Si scriva una procedura che riceva in ingresso un codice di proiezione, una fascia di posti (A, B o C) ed il quantitativo N dei posti che si vogliono occupare. La procedura verifica che vi siano almeno N posti della fascia richiesta nella sala della proiezione e in caso favorevole procede all'assegnazione dei primi posti liberi (i posti si suppongono ordinati per Fila e NumPosto). L'assegnazione di un posto viene effettuata inserendo nella tabella BIGLIETTO una riga con la proiezione e il codice del posto assegnato. Se il numero di posti liberi nella fascia richiesta è insufficiente rispetto alla richiesta viene fornita una segnalazione di errore. (Nell'esercizio si assuma di NON poter usare aspetti di SQL che consentono di limitare ad un numero predefinito il numero di righe restituite da una SELECT).*

Esercizio 24 *Nell'ambito della gestione della concorrenza*

1. *Si dica che cosa si intende quando si dice che uno scheduling è serializzabile;*
2. *Si descriva la tecnica di timestamping per la gestione della concorrenza;*
3. *Si assuma che negli scheduling di seguito riportati vi siano le operazioni di tre transazioni distinte (il pedice qualifica la transazione e corrisponde al timestamp della transazione). Si dica negli scheduling considerati quali transazioni vengono eseguite con successo.*

$S' = R_1(X), R_2(Y), W_1(X), W_2(Y), R_3(Z), R_2(X)$

$S'' = R_1(X), R_2(Y), R_2(X), W_2(Y), W_1(X), R_3(Z)$

$S''' = R_2(X), R_3(X), W_3(X), W_2(X), W_1(Z)$

3 Prova scritta del 01-07-04

Si consideri il seguente schema relazionale che descrive il contenuto di libri:

$LIBRI(ISBN, TITOLO)$

$AUTORI(ISBN, NOME, COGNOME)$

$CAPITOLI(ISBN, N - CAP, PAG - I, PAG - F, TITOLO)$

$FIGURE(ISBN, N - FIG, DIDASCALIA, PAG)$.

$LIBRI$ contiene codice ISBN e titolo dei libri; $AUTORI$ contiene la descrizione del nome e del cognome degli autori di un libro; $CAPITOLI$ contiene numero, titolo, pagina iniziale e finale di ogni capitolo di un libro; $FIGURE$ contiene numero, didascalia e pagina di ogni figura di un libro. (Ovviamente un libro può avere più autori, capitoli e figure.)

Esercizio 31 Senza fare uso delle operazioni di conteggio, si scriva una espressione in algebra relazionale che, se valutata, fornisce i titoli dei libri scritti da un solo autore e senza figure.

Esercizio 32 Scrivere una interrogazione in SQL che restituisca il titolo di libri che abbiano al più una figura per capitolo.

Esercizio 33 Si supponga che nella base di dati sia stata già creata una tabella di schema $TITOLICAPITOLI(ISBN, CONCTITOLI)$. Per un libro identificato dal suo ISBN, l'attributo $CONCTITOLI$ contiene la concatenazione di tutti i titoli dei capitoli (ordinati per numero di capitolo) di quel libro separati da un carattere ','.

Esempio. Se il libro di ISBN '0-55-1' ha i capitoli di titolo 'Introduzione a SQL', 'Sintassi di SQL' e 'Conclusioni', allora l'attributo $CONCTITOLI$ contiene la stringa 'Introduzione a SQL;Sintassi di SQL;Conclusioni'.

Si scriva una procedura in C o Pascal (senza parametri di ingresso) che ha come effetto la popolazione della tabella $TITOLICAPITOLI$ (inserisce una riga per ogni libro presente nella tabella $LIBRI$).

Esercizio 34 Si descriva la struttura di un B^+ albero e si scriva la sequenza di B^+ alberi di grado 3 con chiavi che assumono come valori numeri interi ottenuta, a partire dall'albero vuoto, inserendo la seguente sequenza di valori di chiave: 8, 5, 3, 4, 2, 9, 10.

Soluzione Esercizio 31

Siano LI , AU , CP , FG le relazioni per $LIBRI$, $AUTORI$, $CAPITOLI$, $FIGURE$, rispettivamente.

ISBN di libri con almeno due autori.

$$PIU \leftarrow \Pi_{ISBN}(\rho_{AU1(ISBN1, NOME1, COGNOME1)}(AU) \bowtie_C AU)$$

dove C è la condizione $ISBN = ISBN1 \wedge (COGNOME \neq COGNOME1 \vee NOME \neq NOME1)$.

ISBN di libri con figure.

$$FIG \leftarrow \Pi_{ISBN}(FI)$$

Risultato.

$$\Pi_{TITOLO}(LI \bowtie (\Pi_{ISBN}(LI) \setminus (PIU \cup FIG)))$$

† **Soluzione Esercizio 32**

```
SELECT TITOLO
FROM  LIBRI AS
WHERE ISBN NOT IN
      (SELECT C.ISBN
       FROM  (CAPITOLI AS C JOIN FIGURE AS F1 ON C.ISBN=F1.ISBN)
              JOIN FIGURE AS F2 ON C.ISBN=F2.ISBN
       WHERE F1.N-FIG <> F2.N-FIG AND F1.PAG BETWEEN C.PAG-I AND C.PAG-F
              AND F2.PAG BETWEEN C.PAG-I AND C.PAG-F)
```

†
Soluzione Esercizio 33

PROCEDURE Scarica

```
VAR
EXEC SQL BEGIN DECLARE SECTION
      Lib, Cap, StTit, CurrLib, Conc: string;
      NCap : integer
EXEC SQL END DECLARE SECTION

BEGIN
EXEC SQL DECLARE scan CURSOR FOR
      SELECT ISBN, N-CAP, TITOLO
      FROM  CAPITOLI
      ORDER BY ISBN, N-CAP;
EXEC SQL OPEN scan;
EXEC SQL FETCH scan INTO :Lib, :Cap, :StTit;
IF SQLCODE = 0 THEN
      BEGIN  CurrLib := Lib;
      Conc := StTit
      END
WHILE SQLCODE = 0 DO
      BEGIN
      EXEC SQL FETCH scan INTO :Lib, :Cap, :StTit;
      WHILE SQLCODE = 0 AND CurrLib = Lib DO
              BEGIN
              Conc := Conc + ';' + StTit;
              EXEC SQL FETCH scan INTO :Lib, :Cap, :StTit;
              END
      EXEC SQL INSERT INTO TITOLICAPITOLI VALUES (:CurrLib, :Conc)
      END
END
```

†