

Esame Basi di Dati 26/09/2014
Prof. Alfredo Pulvirenti

- 1) Rispondere alle seguenti domande. 2 punti se la risposta è giusta, -2 se la risposta è sbagliata.
- a) Sono date due relazioni $R(A,B)$ e $S(B,C)$ dove B è chiave primaria in S . La cardinalità di una relazione è indicata col simbolo $\#$. Le cardinalità delle due relazioni sono: $\#(R) = 5000$ e $\#(S) = 2000$.
Quindi: $\#(R \text{ JoinNaturale } S) \leq ?$
- b) B : Sono date due relazioni $R(A,B,C)$ e $S(A,D,E)$ dove A è chiave primaria in S e $R.A$ è una chiave esterna su $S.A$.
Quindi: $\#(R \text{ JoinNaturale } S) = \#(R)$. VERO o FALSO?

2. Algebra ed SQL

Dato lo schema di database:

TAXI(id_auto, targa, modello, km_percorsi)

TASSISTA(id_tass, nome, cognome, n_patente, eta, data_assunzione)

TURNO(id_turno, id_tassista, id_auto, data, ora_inizio, ora_fine)

CHIAMATA(id_chiamata, luogo_partenza, luogo_destinazione, km, id_turno, data_ora)

1. Definire le chiavi primarie e le chiavi esterne delle tabelle [0 punti esatta, -2 errata]
2. Implementare le seguenti query in algebra relazionale:
 - a. [1 punto] Trovare i tassisti, restituendo numero patente, nome e cognome, assunti dopo il 10/1/2014 che hanno effettuato corse con luogo di partenza "Via Etnea" con auto di modello "Fiat Multipla".
 - b. [2 punti] Trovare i tassisti che hanno guidato tutti i taxi.
 - c. [2 punti] Tra le auto che hanno effettuato corse in "Via Roma" (come partenza o destinazione), trovare quello che hanno percorso più chilometri. Restituire targa e modello.
3. Implementare le seguenti query SQL:
 - a. [1 punto] Trovare il numero di chiamate che ci sono state per ogni turno di giorno 31-12-2013.
 - b. [2 punti] Trovare le coppie di tassisti che non hanno mai guidato lo stesso taxi lo stesso giorno.
 - c. [2 punti] Trovare il tassista che ha percorso più chilometri in assoluto.

3. Progettazione

Si vuole progettare un database per rappresentare in maniera strutturata la natura ipertestuale di un insieme di documenti. Ogni documento è rappresentato da un titolo, una categoria, una data di pubblicazione, un autore e un insieme di parole che compongono il corpo dell'articolo stesso. Una certa parola in un documento può fare quindi riferimento ad un altro documento (hyperlink). Il tutto assomiglia molto alla struttura che tutti conoscete sul Web.

- a) [7 punti] Progettate il tutto indicando quindi lo schema ER, indicate se lo schema è normalizzato o meno, lo schema relazionale e tutti i vincoli relativi.
- b) Durante la vostra progettazione tenete in considerazione di dover rispondere alla seguente query in SQL sullo schema che definirete
[3 punti] Le prime due parole più comuni tra tutti i documenti.

XML

Dato il seguente DTD:

```
<!ELEMENT collezione (descrizione,ricette*)>
<!ELEMENT descrizione ANY>
<!ELEMENT ricetta (titolo,ingredienti*,preparazione,commenti?,fattori_nutritivi)>
<!ELEMENT titolo (#PCDATA)>
<!ELEMENT ingredienti (ingrediente*,preparazione)?>
<!--ATTLIST ingredienti nome CDATA #REQUIRED
           quantità CDATA #IMPLIED
           unità CDATA #IMPLIED-->
<!ELEMENT preparazione (passi*)>
<!ELEMENT passo (#PCDATA)>
<!ELEMENT commenti (#PCDATA)>
<!ELEMENT fattori_nutritivi EMPTY>
<!--ATTLIST fattori_nutritivi proteine CDATA #REQUIRED
           carboidrati CDATA #REQUIRED
           grassi CDATA #REQUIRED
           calorie CDATA #REQUIRED
           alcol CDATA #IMPLIED-->
```

- (a) [2 punti] Dare un file xml che descriva una ricetta con 3 passi.
- (b) [4 punti] Scrivere una query in xquery per trovare le ricette che hanno 450 calorie e che richiedono 5 passi