

Cognome e nome: _____ **Matricola:** _____ **Turno:** _____

Riportare sui fogli i seguenti dati: cognome, nome, matricola e turno di laboratorio.

Esame di SQL

Punteggi massimi:

- Domande 1 e 2 svolte perfettamente: 23;
- Domande 1 e 3 svolte perfettamente: 25;
- Domande 2 e 3 svolte perfettamente: 28;
- Domande 1, 2 e 3 svolte perfettamente: 33.

Lo svolgimento corretto di una sola domanda non permette il raggiungimento della sufficienza.

Le seguenti relazioni definiscono una base di dati **“Presenze”** per gestire le informazioni sulla frequenza delle lezioni da parte di studenti e studentesse. Gli attributi sottolineati sono le chiavi primarie delle relazioni.

STUDENTE(Matricola, Cognome, Nome, DataNascita, TipoStudente*)

INSEGNAMENTO(Codice, Titolo, CFU)

DOCENTE(Matricola, Cognome, Nome, Qualifica)

LEZIONE(Data, Ora, Aula, Insegnamento, Docente)

PARTECIPAZIONE(Data, Ora, Aula, Studente, Modalità)

LEZIONE(Insegnamento) referencia INSEGNAMENTO(Codice),

LEZIONE(Docente) referencia DOCENTE(Matricola),

PARTECIPAZIONE(Data, Ora, Aula) referencia LEZIONE(Data, Ora, Aula),

PARTECIPAZIONE(Studente) referencia STUDENTE(Matricola).

“TipoStudente” assume il valore ‘Occupat*’ se lo studente o la studentessa è lavoratore/lavoratrice, altrimenti assume il valore ‘Non occupat*’. Questo attributo può assumere valore NULL.

“Qualifica” può assumere i seguenti valori ‘RU’ (Ricercatore/ricercatrice a tempo indeterminato), ‘RTD’ (Ricercatore/ricercatrice a tempo determinato), ‘PA’ (Professore/Professoressa Associat*), ‘PO’ (Professore/Professoressa ordinari*), ‘Contratto’ (docente a contratto).

Modalità può assumere i valori ‘Presenza’ (se la persona ha partecipato in presenza alla lezione), ‘Online’ (se la persona ha partecipato a distanza alla lezione).

Tutti gli altri attributi sono autoesplicativi.

Domanda 1 (bassa complessità).

Elencare codice e titolo degli insegnamenti seguiti anche solo parzialmente a distanza da studenti e studentesse non occupat* o di tipo sconosciuto. Ordinare i risultati per numero decrescente di CFU e, successivamente, per titolo dell’insegnamento crescente.

Soluzione 1.

```
SELECT DISTINCT i.Codice, i.Titolo
FROM studente s JOIN partecipazione p ON (p.Studente=s.Matricola)
      JOIN lezione l ON (l.Data=p.Data AND l.Ora=p.Ora AND l.Aula=p.Aula)
      JOIN insegnamento i ON (l.Insegnamento=i.Codice)
WHERE Modalità='Online' AND (TipoStudente = 'Non occupat*' OR TipoStudente IS NULL)
ORDER BY i.CFU DESC, i.Titolo ASC;
```

Domanda 2 (media complessità).

Per ogni docente non professore o professoressa, elencare i titoli degli insegnamenti aventi almeno una lezione con più di 50 partecipanti non occupat* in presenza.

Soluzione 2.

```
SELECT DISTINCT d.Matricola, i.Titolo
FROM studente s JOIN partecipazione p ON (p.Studente=s.Matricola)
      JOIN lezione l ON (l.Data=p.Data AND l.Ora=p.Ora AND l.Aula=p.Aula)
      JOIN insegnamento i ON (l.Insegnamento=i.Codice)
      JOIN docente d ON (d.Matricola=l.Docente)
WHERE s.TipoStudente='Non occupat*' AND p.Modalità='Presenza' AND d.Qualifica<>'PO'
AND d.Qualifica<>'PA'
GROUP BY d.Matricola, i.Codice, p.Data, p.Ora, p.Aula
HAVING COUNT(*) > 50;
```

Domanda 3 (alta complessità).

Per ogni professoressa o professore ordinario, mostrare il numero di insegnamenti in cui il numero di partecipanti in presenza supera sempre il numero di partecipanti a distanza.

Soluzione 3.

```
WITH NumPartecipantiDistanza AS (
  SELECT Data, Ora, Aula, COUNT(*) AS NumPartDist
  FROM partecipazione
  WHERE Modalità='Online'
  GROUP BY Data, Ora, Aula
), NumPartecipantiPresenza AS (
  SELECT Data, Ora, Aula, COUNT(*) AS NumPartPres
  FROM partecipazione
  WHERE Modalità='Presenza'
  GROUP BY Data, Ora, Aula
)
SELECT d.Matricola, d.Cognome, d.Nome, COUNT(DISTINCT Insegnamento)
FROM docente d JOIN lezione l ON (l.Docente=d.Matricola)
WHERE d.Qualifica='PO' AND l.Insegnamento NOT IN (
  SELECT Insegnamento
  FROM lezione l JOIN NumPartecipantiPresenza npp ON
    (npp.Data=l.Data AND npp.Ora=l.Ora AND npp.Aula=l.Aula)
  JOIN NumPartecipantiDistanza npd ON
    (npd.Data=l.Data AND npd.Ora=l.Ora AND npd.Aula=l.Aula)
  WHERE NumPartPres<=NumPartDist )
GROUP BY d.Matricola;
```

Esame di Teoria

Domanda 1 (9 punti).

Con riferimento alla base di dati “**Presenze**”:

- A. (5 punti) Esprimere in Algebra Relazionale l'interrogazione
Elencare le matricole degli studenti/delle studentesse che hanno partecipato in presenza a tutte le lezioni del corso di Basi di Dati.
- B. (4 punti) Esprimere, nel calcolo relazionale su tuple con dichiarazione di range, la seguente domanda:
Elencare Titolo e numero di CFU degli insegnamenti tenuti da almeno due docenti.

Soluzione 1.

A. Una possibile soluzione è la seguente:

$$\pi_{Matricola, Data, Ora, Aula} \left(\sigma_{Modalità='Presenza'} (partecipazione \bowtie_{\alpha} lezione \bowtie_{\beta} insegnamento) \right) \\ \div \pi_{Data, Ora, Aula} (\sigma_{Insegnamento='BD'} (lezione))$$

dove

$$\alpha = PARTECIPAZIONE.Data = LEZIONE.Data \wedge PARTECIPAZIONE.Ora \\ = LEZIONE.Ora \wedge PARTECIPAZIONE.Aula = LEZIONE.Aula \\ \beta = LEZIONE.Insegnamento = INSEGNAMENTO.Codice$$

B. Una possibile soluzione è la seguente:

$$\{i.Titolo, i.CFU \mid i(INSEGNAMENTO) \mid \exists l(LEZIONE)(l.Insegnamento=i.Codice \wedge \\ \exists l'(LEZIONE)(l'.Insegnamento=i.Codice \wedge l.Docente \neq l'.Docente))\}$$

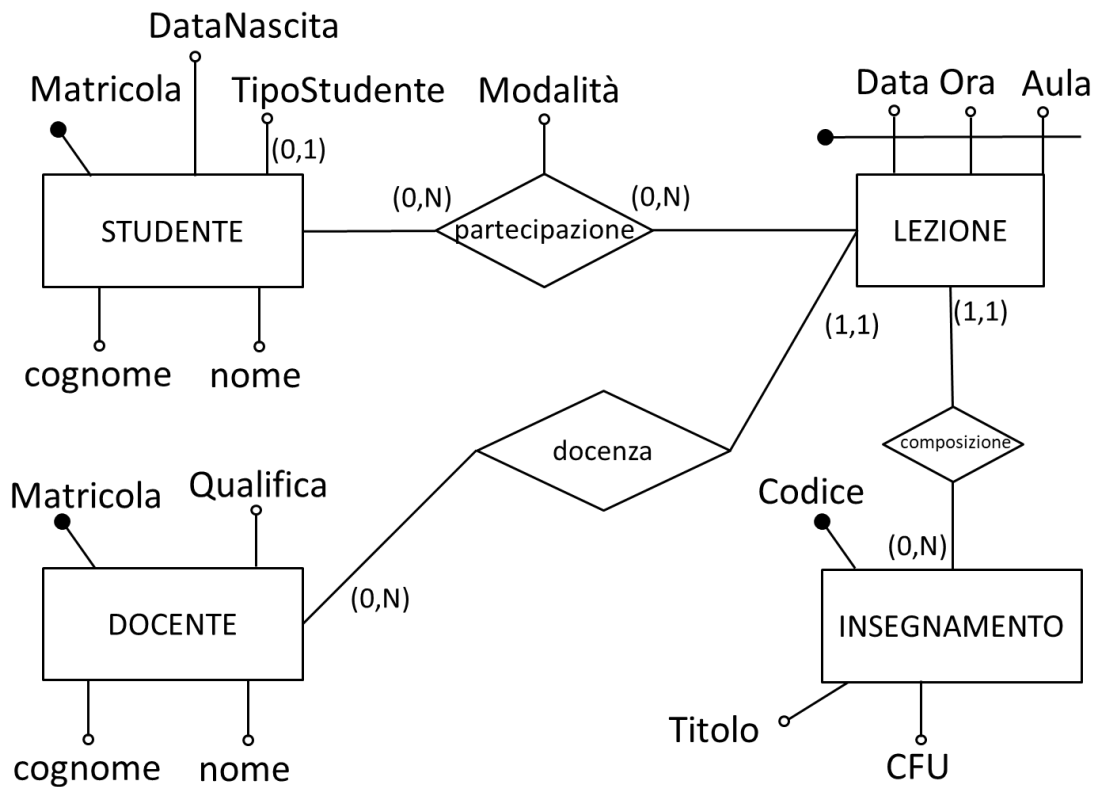
Domanda 2 (9 punti).

- A. Indicare la condizione necessaria e sufficiente affinché una decomposizione di una relazione $R(A)$ in $R_1(A_1)$ e $R_2(A_2)$ sia senza perdita di informazione.
- B. Disegnare uno schema Entity Relationship che potrebbe aver generato lo schema logico della base di dati “Presenze”.
- C. Ipotizzare una dipendenza funzionale **sensata(*)** che renda lo schema “Presenze” in 3NF (terza forma normale) ma non in BCNF (Forma normale di Boyce-Codd).
(*) cioè, con un significato giustificabile da un ipotetico sistema informativo rappresentato dallo schema “Presenze”.

Soluzione 2.

A. Si vedano gli appunti/testo/slide.

B. Una possibile soluzione è la seguente:



C. La dipendenza deve essere per forza del tipo “attributi primi” e le uniche possibili sono:

Insegnamento \rightarrow Aula (in Lezione)

Docente \rightarrow Aula (in Lezione)

Altre dipendenze sono difficilmente giustificabili dal punto di vista del sistema informativo.

Domanda 3 (7 punti).

Con riferimento alla base di dati “**Presenze**” e i seguenti dati quantitativi:

CARD(DOCENTE) = 100

CARD(LEZIONE) = 5000

VAL(Insegnamento, LEZIONE) = 50

VAL(Aula, LEZIONE) = 10

disegnare gli alberi sintattici prima e dopo l'ottimizzazione logica e calcolare il numero di tuple “mosse” prima e dopo l'ottimizzazione logica della seguente query:

$\sigma_{(Qualifica \neq 'PO' \wedge Insegnamento='BD' \wedge Aula='D')} (lezione \bowtie_{Docente=Matricola} docente)$

Soluzione 3.

La query ottimizzata dividendo la selezione e portandola verso le foglie è:

$(\sigma_{Qualifica \neq 'PO'} (docente))$

$\bowtie_{Docente=Matricola}$

$(\sigma_{Insegnamento='BD' \wedge Aula='D'} (lezione))$

Prima dell'ottimizzazione:

- Costo $r1 = (\text{lezione} \bowtie_{\text{Docente=Matricola}} \text{docente}): \text{CARD}(\text{LEZIONE}) \times \text{CARD}(\text{DOCENTE}) = 5000 \times 100 = 5 \times 10^5$.
- Cardinalità $|r1| = \text{CARD}(\text{LEZIONE}) = 5000$ (equi-join attraverso la chiave esterna)
- Costo della selezione: $|r1| = 5000$
- Costo totale = $500\,000 + 5000 \sim 5 \times 10^5$.

Dopo l'ottimizzazione:

- Costo $\sigma_1 = \sigma_{\text{Qualifica} \neq \text{'PO'}}(\text{docente}) = \text{CARD}(\text{DOCENTE}) = 100$
- Costo $\sigma_2 = \sigma_{(\text{Insegnamento} = \text{'BD'} \wedge \text{Aula} = \text{'D'})}(\text{lezione}) = \text{CARD}(\text{LEZIONE}) = 5000$
- Tuple prodotte dalla selezione $|\sigma_1| = (1 - 1/\text{VAL}(\text{Qualifica}, \text{DOCENTE})) \times \text{CARD}(\text{DOCENTE}) = (1 - 1/5) \times 100 = 80$
- Tuple prodotte dalla selezione $|\sigma_2| = f_{\text{Insegnamento} = \text{'BD'}} \times f_{\text{Aula} = \text{'D'}} \times \text{CARD}(\text{LEZIONE}) = 1/\text{VAL}(\text{Insegnamento}, \text{LEZIONE}) \times 1/\text{VAL}(\text{Aula}, \text{LEZIONE}) \times \text{CARD}(\text{LEZIONE}) = (1/50) \times (1/10) \times 5000 = 10$
- Costo join $r = \sigma_1 \bowtie_{\text{Docente=Matricola}} \sigma_2 = 80 \times 10 = 800$.
- Costo totale = $100 + 5000 + 800 = 5900 \sim 6 \times 10^3$.

Domanda 4 (8 punti).

- Enunciare il criterio di serializzabilità.
- Dire, giustificando la risposta, se la seguente storia è compatibile con il protocollo 2PL (lock a due fasi):
 $S = r1(x), r2(x), w1(z), w2(y), w3(x), r1(y), w2(z)$

Soluzione 4.

- Si vedano gli appunti/slide.
- Non è compatibile perché la transazione T1 sarebbe costretta a riacquisire il lock su y dopo aver rilasciato quello su x per permettere a T3 di acquisire a sua volta il lock su x. Inoltre, T1 non può acquisire prima il lock su y, perché dovrebbe poi lasciarlo a T2.