



UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Corso di Laurea in Informatica

Esame di Basi di Dati

Esame del 30 Gennaio 2020

Regole dell'esame:

- Il tempo massimo per l'esame è di tre ore, dalle 9.30 alle 12.30.
- Non è possibile utilizzare né appunti, né il libro, né altro materiale.
- **Inserire le risposte nei riquadri che seguono i testi degli esercizi**
- L'uso della matita per mostrare la soluzione degli esercizi è fatta a rischio e pericolo degli studenti. Nessuna rivendicazione verrà accettata in caso parte della soluzione si cancelli, per via delle caratteristiche transitorie dei tratti a matita.
- Occorre avere acquisito almeno metà dei punti tra Esercizio 1 e Esercizio 2 (cioè almeno 7.5 punti).
- Occorre avere acquisito almeno metà dei punti dell'Esercizio 3 (cioè almeno 3.5 punti).
- **Coloro che siano sorpresi (1) a copiare, (2) a far copiare, (3) ad usare il cellulare o (4) a consultare qualsiasi materiale:**
 - Dovranno lasciare l'aula dell'esame;
 - Riceveranno 0 punti "di ufficio";
 - Dovranno saltare l'appello successivo.

Nome: _____ Num. Matric.: _____

Esercizio 1: Diagramma ER (8 punti)

Si richiede di produrre lo schema concettuale Entità-Relazione di un'applicazione relativa ad una ditta che estrae marmo per abitazioni da cave, evitando di introdurre entità non necessarie. Disegnare il diagramma ER nel riquadro della pagina che segue.

Si vuole progettare lo schema ER di una base di dati che supporti la gestione di riviste scientifiche e degli articoli pubblicati in esse.

Una rivista scientifica è caratterizzata da un nome univoco, dall'anno di fondazione e dalla tematica trattata (e.g., Computer Science, Architettura, ecc.).

Ciascuna rivista prevede delle edizioni mensili, identificate da un codice univoco e caratterizzate da mese, anno di pubblicazione e prezzo. Ogni edizione mensile è strutturata in pagine (almeno una pagina è sempre presente); una pagina è identificata da una numerazione univoca all'interno dell'edizione.

Ogni articolo è pubblicato in un'unica edizione ed è identificato da un codice univoco all'interno di tale edizione. Di un articolo si conoscono titolo e testo. Ogni articolo occupa un certo numero di pagine (almeno una).

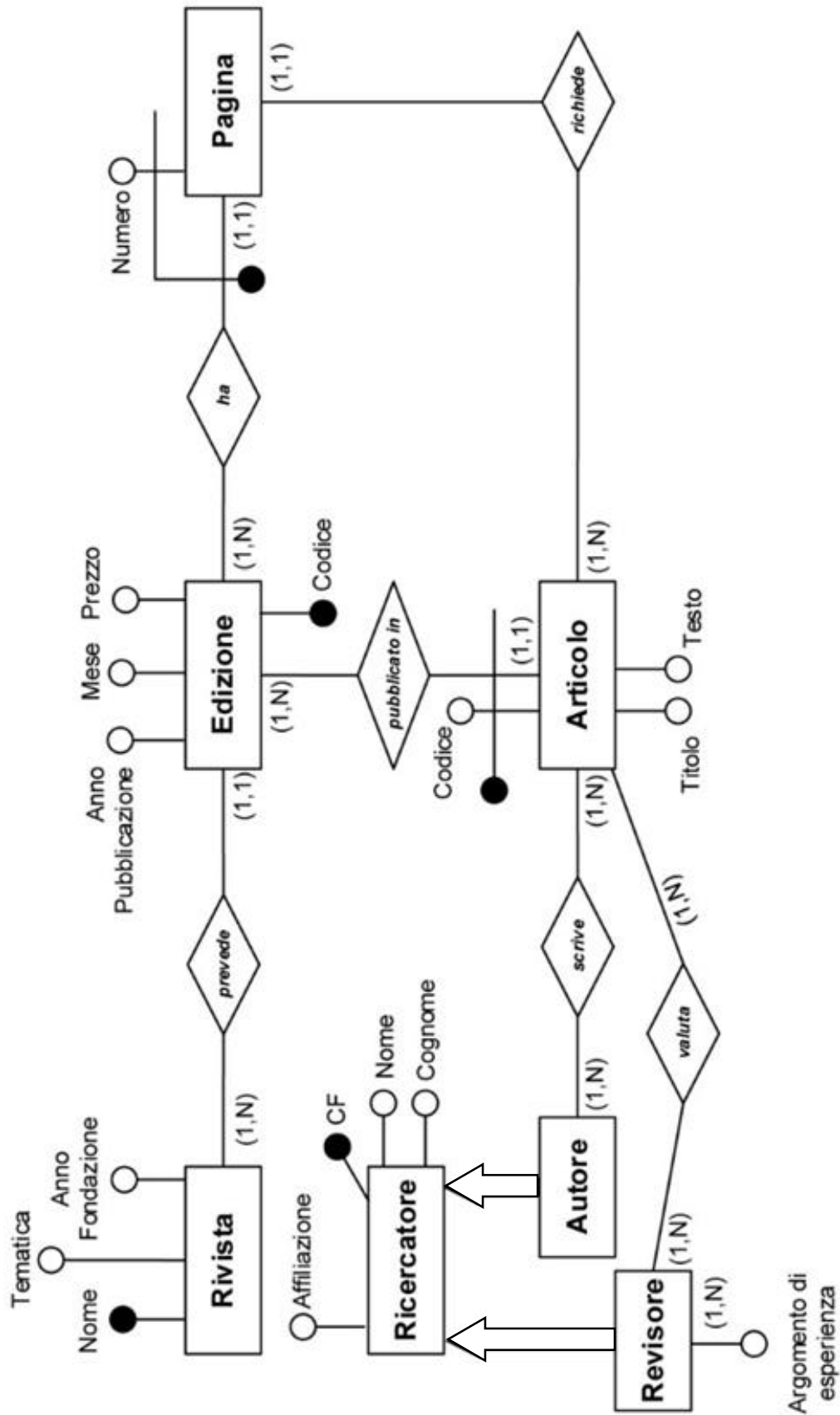
Nella pubblicazione di articoli in riviste scientifiche, sono coinvolti ricercatori. Di ogni ricercatore, si conoscono nome, cognome, codice fiscale, e l'università o centro di ricerca presso la quale afferiscono.

In particolare, alcuni ricercatori sono anche revisori in quanto valutano la qualità degli articoli. Si vuole memorizzare, per ogni articolo, quali revisori (almeno uno) lo hanno revisionato. I ricercatori che sono anche revisori sono associati con un certo numero di argomenti di esperienza.

I ricercatori sono anche autori nel senso che possono pubblicare articoli: per ogni articolo, si vogliono conoscere quali ricercatori ne sono stati autore.

Si noti che un revisore può essere anche un autore, ed un autore può essere anche revisore. Tuttavia, è possibile che un revisore non sia autore, o viceversa.

Soluzione Esercizio 1



Esercizio 2: Progettazione Logica (7 punti)

A partire dallo ER concettuale dell'Esercizio 1, produrre uno schema relazionale del database nel riquadro sottostante, minimizzando i valori nulli. Indicare la chiave primaria, i vincoli di chiave esterne e gli attributi che ammettono valori nulli. Illustrare come ristrutturare il diagramma ER per essere direttamente traducibile in uno schema relazionale.¹

Soluzione Esercizio 2

Nella ristrutturazione:

- l'entità *Ricercatore* viene rimossa e i suoi attributi duplicati per le due entità figlie.
- l'attributo multivalore *Argomento di esperienza* viene tradotto come segue dove l'entità *ARGOMENTO* contiene un solo attributo *Argomento*, che funge da identificato:



RIVISTA (Nome, Tematica, Anno)

EDIZIONE(Codice, Anno, Mese, Prezzo, NomeRivista)

- Chiave Esterna: EDIZIONE.NomeRivista → RIVISTA.Nome

ARTICOLO(Codice, Edizione, Titolo, Testo)

- Chiave Esterna: ARTICOLO.Edizione → EDIZIONE.Codice

PAGINA(Numero, Edizione, CodiceArt)

- Chiave Esterna: PAGINA.Edizione → EDIZIONE.Codice
- Chiave Esterna: PAGINA.(Edizione,CodiceArt) → ARTICOLO.(Edizione,Codice)

AUTORE(CF, Affiliazione, Nome, Cognome)

REVISORE(CF, Affiliazione, Nome, Cognome)

SCRIVE(Autore, Codice, Edizione)

- Chiave Esterna: SCRIVE.Autore → Autore.CF
- Chiave Esterna: SCRIVE.(Codice, Edizione) → ARTICOLO.(Codice, Edizione)

VALUTA(Revisore, Codice, Edizione)

- Chiave Esterna: VALUTA.Revisore → REVISORE.CF
- Chiave Esterna: VALUTA.(Codice, Edizione) → ARTICOLO.(Codice, Edizione)

ARGOMENTO(Argomento)

ESPERIENZA(Revisore, Argomento)

- Chiave Esterna: ESPERIENZA.Argomento → ARGOMENTO.Argomento
- Chiave Esterna: ESPERIENZA.Revisore → REVISORE.CF

¹ Allo scopo di mostrare la ristrutturazione del diagramma ER, è possibile semplicemente mostrare i cambiamenti apportati al diagramma nel riquadro soluzione dell'Esercizio 1, utilizzando una penna di diverso colore (non rossa!)

Esercizio 3: Algebra Relazionale & SQL (7 punti)

Si consideri la seguente base di dati per la registrazione degli impiegati che lavorano in aziende:

- IMPIEGATO(CF, Nome, Cognome, Dipartimento, Stipendio)
- LAVORA(CF, PIVA, DataInizio, DataFine)
- AZIENDA(PIVA, Citta, Nazione, Continente)

dove DataFine può essere NULL se l'impiegato lavora ancora in una azienda

- A. Nel riquadro, scrivere una Query in Algebra Relazione che restituisce il codice fiscale degli impiegati che hanno lavorato in almeno due nazioni differenti. (2 punti).²

LA1 = AZIENDA \bowtie LAVORA

LA2 = AZIENDA \bowtie LAVORA

$\pi_{LA1.CF}$

(LA1 $\bowtie_{LA1.CF = LA2.CF \text{ AND } LA1.Nazione \neq LA2.Nazione}$ LA2)

² Si assuma che l'operatore di join $A \bowtie B$ **senza condizioni** mantenga le tuple di $A \times B$ con valori uguali su attributi uguali (join naturale). Se **una condizione C è specificata**, \bowtie_C mantiene le tuple di $A \times B$ per cui la condizione C è vera.

Nome: _____ Num. Matric.: _____

- B. Nel riquadro, scrivere una query SQL che restituisca il codice fiscale delle persone che attualmente lavorano in esattamente un'azienda, cioè escludendo anche coloro che attualmente non lavorano. Ogni CF può apparire al più una volta (2.5 punti).

```
SELECT DISTINCT CF
FROM LAVORA
WHERE DATAFINE IS NOT NULL
EXCEPT
SELECT L1.CF
FROM LAVORA L1, LAVORA L2
WHERE

    L1.MATR=L2.MATR AND L1.PIVA<>L2.PIVA
    AND L1.DATAFINE IS NULL
    AND L2.DATAFINE IS NULL
```

- C. Nel riquadro, scrivere una query SQL che restituisce il cognome degli impiegati che lavorano o hanno lavorato in tutte le aziende. (2.5 punti)

Una soluzione giusta è:

```
CREATE VIEW CF-LAVORA-OVUNQUE (CF) AS
SELECT CF
FROM (SELECT DISTINCT CF, PIVA FROM LAVORA)
GROUP BY CF
HAVING COUNT(*) =

    (SELECT COUNT(*) FROM AZIENDA )

SELECT COGNOME
FROM IMPIEGATO I, CF-LAVORA-OVUNQUE C
WHERE I.CF=C.CF
```

Tuttavia per l'esame va bene se in questa Query si assume che ogni persona non possa lavorare più di una volta presso una stessa azienda:

```
CREATE VIEW CF-LAVORA-OVUNQUE (CF) AS
SELECT CF FROM LAVORA GROUP BY CF
HAVING COUNT(*) = (SELECT COUNT(*) FROM AZIENDA )
```

Esercizio 4: Normalizzazione (6 punti)

Data la relazione $R(A,B,C,D)$ con dipendenze funzionali $\{ C \rightarrow D, C \rightarrow A, B \rightarrow C \}$.

1. Mostrare tutte le chiavi di R e motivare perché ognuna è chiave.
2. Dire quale dipendenze violano la forma normale di Boyce Codd (BCNF), spiegandone la ragione.
3. Decomporre in BCNF

1. $B^+ = \{B, C, A, D\}$ e $C^+ = \{C, A, D\}$. Quindi, B è chiave perché la sua chiusura contiene tutti gli attributi della relazione
2. $C \rightarrow D$ e $C \rightarrow A$ violano BCNF perché C non è una superchiave della relazione
3. È possibile decomporre come segue: $R_1(\underline{C}, D)$, $R_2(\underline{C}, A)$ e $R_3(\underline{B}, C)$. Eventualmente – non necessario per l'esame – è possibile rimettere insieme R_1 e R_2 , ottenendo la decomposizione: $R_4(\underline{C}, A, D)$, $R_3(\underline{B}, C)$.

Nome: _____ Num. Matric.: _____

Esercizio 5: Quiz (5 punti)

Rispondere alle seguenti domande, sottolineando quale risposta è corretta (solo una è corretta).

Domanda 1 (1.5 Punti)

Data la query `SELECT * FROM R WHERE C='Valore'` sulla relazione $R(\underline{A}, B, C)$. Quale dei seguenti indici in genere assicura le migliori performance in termini di velocità dell'esecuzione della query?

- (1) Indice Hash su C;
- (2) Indice B-Tree su C;³
- (3) Indice Hash su A, B, C;
- (4) Indice B-Tree su A, B, C.

Domanda 2 (1.5 Punti)

Date una qualsiasi istanza della relazioni $R(\underline{A}, B, C)$ e $S(\underline{D}, E, F)$ dove (1) le uniche chiavi di R e S sono rispettivamente $\{A\}$ e $\{D\}$, (2) non ci sono chiavi esterne. Indicato con $|X|$ il numero di tuple di una relazione X, quale è vera tra le seguenti affermazioni relative al numero di tuple in $R \bowtie_{B=F} S$?

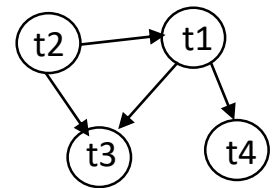
- (1) $0 \leq |R \bowtie_{B=F} S| \leq |R|$
- (2) $0 \leq |R \bowtie_{B=F} S| \leq |R| * |S|$
- (3) $|R \bowtie_{B=F} S| = |R| * |S|$
- (4) $|R| \leq |R \bowtie_{B=F} S| \leq |R| * |S|$

³ Nota sull'esercizio: la risposta "B-Tree su C" non è del tutto sbagliata perché comunque quell'indice velocizza la query, anche se non tanto quanto quello Hash.

Domanda 3 (2 Punti)

Dato il seguente schedule, con grafo di conflitti in figura:

R₂(A) R₂(B) W₁(A) W₂(D) R₃(C) R₁(C) W₃(B) R₄(A) W₃(C).



Quale delle seguenti affermazioni è vera?

- (1) S è conflict-serializzabile ma **non** view-serializzabile
- (2) S **non** è conflict-serializzabile ma è view-serializzabile
- (3) S è sia conflict-serializzabile che view-serializzabile⁴
- (4) S **non** è né conflict-serializzabile né view-serializzabile

⁴ Nota sulla soluzione della Domanda 3: non è necessario verificare la view-serializzabilità perché la teoria dice che, se lo schedule è conflict-serializzabile, allora è anche view-serializzabile.