



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

# Corso di Laurea in Informatica

## Esame di Basi di Dati

Esempio di Esame

Regole dell'esame:

- Non è possibile utilizzare alcun materiale, né appunti, né il libro.
- **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 8 punti).
- Occorre avere acquisito almeno metà dei punti dell'Esercizio 3 (cioè 3 punti, arrotondando all'intero più piccolo).
- **Coloro che siano sorpresi a copiare o a far copiare:**
  - Dovranno lasciare l'aula dell'esame
  - Riceveranno 0 punti "di ufficio"
  - Dovranno saltare l'appello successivo

Nome: \_\_\_\_\_ Num. Matric.: \_\_\_\_\_

### **Esercizio 1: Diagramma ER (9 punti)**

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

Di ogni blocco di marmo estratto interessa il codice (identificativo), l'anno di estrazione, il peso e la cava da cui è stato estratto.

Per il dominio di estrazione del marmo, è interesse memorizzare i dati di certi luoghi geografici. Ogni luogo geografico ha un codice identificativo e l'area che occupa in chilometri quadrati.

Le cave di interesse sono quelle dalle quali è stato estratto almeno un blocco. Ogni cava è un luogo geografico di cui interessa anche conoscere l'altitudine e la regione (esattamente una) in cui è situato il suo territorio.

Da ogni blocco di marmo si producono almeno una lastra di marmo e di ogni lastra interessa il blocco da cui è stato prodotto (uno ed uno solo), la superficie, il numero (unico nell'ambito del blocco di marmo da cui è stato prodotto), l'eventuale abitazione in cui viene usata.

Ogni abitazione che interessa all'applicazione usa almeno una lastra di marmo e di ognuna di tali abitazioni interessa il codice (identificativo), la classe e le eventuali banche che hanno partecipato all'asta per quell'abitazione, con la data di partecipazione della banca all'asta.

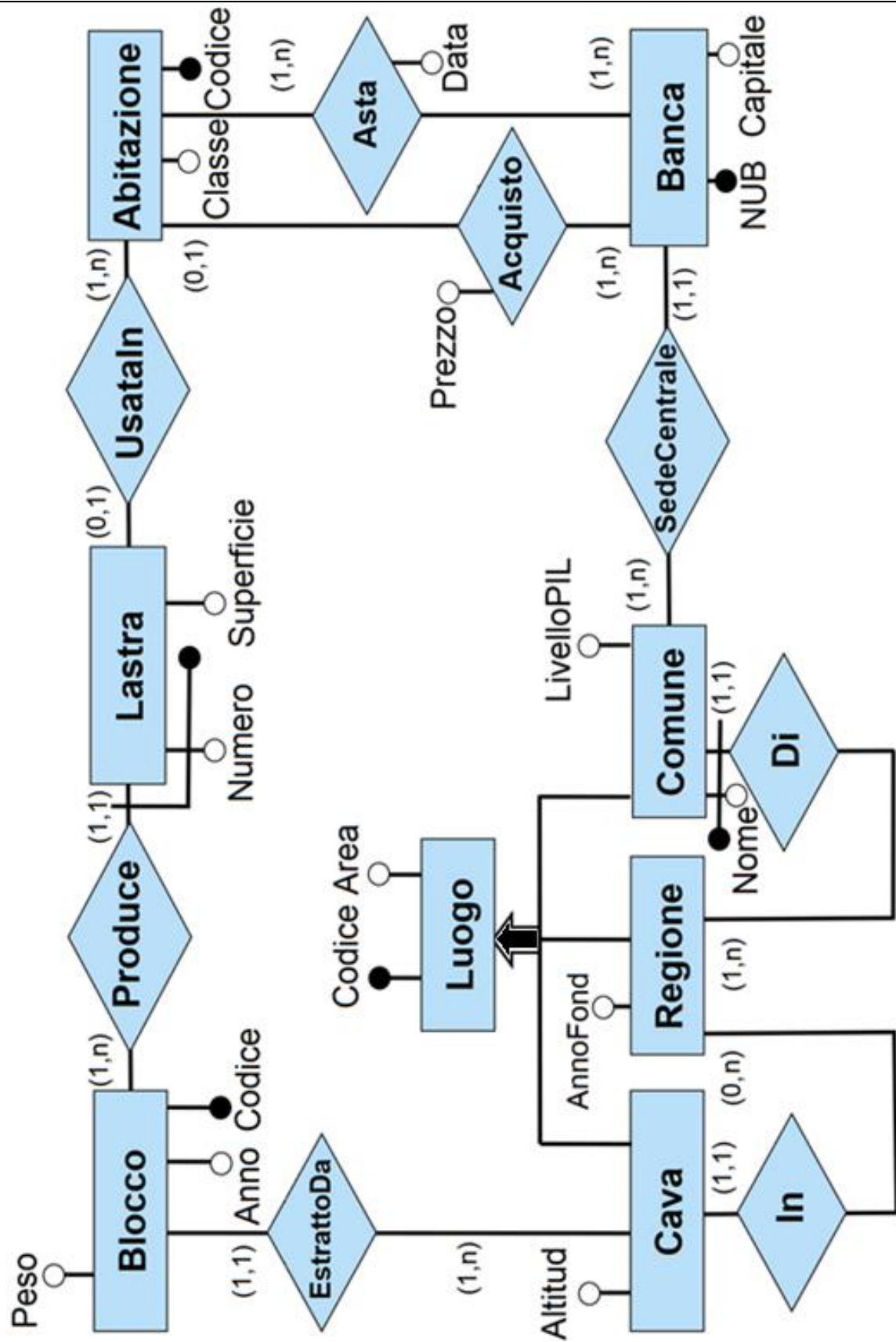
Tra le banche che hanno partecipato all'asta per una certa abitazione, dopo tale asta, è di interesse sapere la banca, se esiste, che ha acquistato l'abitazione stessa, con il relativo prezzo di acquisto.

Di ogni banca interessa il "codice unico bancario" (identificativo), il capitale sociale ed il comune in cui si trova la sede centrale.

Di ogni comune, oltre alle proprietà di tutti i luoghi geografici, interessa il livello del PIL, la regione in cui si trova ed il nome (unico nell'ambito della regione in cui si trova).

Di ogni regione, oltre alle proprietà di tutti i luoghi geografici, interessa l'anno della sua fondazione.

## Soluzione Esercizio 1



Nome: \_\_\_\_\_ Num. Matric.: \_\_\_\_\_

## Esercizio 2: Progettazione Concettuale (7 punti)

A partire dallo ER concettuale al punto 1, produrre uno schema relazionale del database nel riquadro sottostante. Indicare la chiave primaria, i vincoli di chiave esterne, quali attributi ammettono valori nulli, ed ogni altro vincoli rilevante. Mostrare il diagramma ER ristrutturato per eliminare i costrutti non direttamente rappresentabili.<sup>1</sup>

### Soluzione Esercizio 2

L'entità **Luogo** viene eliminata, insieme alla generalizzazione. Gli attributi **Codice** e **Area** vengono replicati in **Cava**, **Regione** e **Comune**.

Tutti gli attributi seguiti da un asterisco ammettono valori nulli (Nota: all'esame è possibile rappresentare i valori nulli in qualsiasi modo perché non c'è uno standard. L'importante è che sia chiaro quale vengono rappresentati come nulli)

Regione(Codice, Area, AnnoFondazione)

Cava(Codice, Area, Altitudine, CodRegione)

- Foreign key: Cava.CodRegione → Regione.Codice

Comune(Codice, CodRegione, Nome, Area, LivelloPIL)

- Foreign key: Comune.CodRegione → Regione.Codice
- (CodRegione, Nome) è una seconda chiave.

Banca(NUB, Capitale, CodRegSedeCentrale, NomeComuneSedeCentrale)

- Foreign key:  
Banca.(CodRegSedeCentrale, NomeComuneSedeCentrale) →  
Comune(CodRegione, Nome)

Abitazione(Codice, Classe, BancaAcquisto\*, PrezzoAcquisto\*)

- Foreign key: Abitazione.BancaAcquisto → Banca.NUB

Asta(NUB-Banca, CodiceAbitazione, Data)

- Foreign key: Asta. NUB-Banca → Banca.NUB
- Foreign key: Asta.CodiceAbitazione → Abitazione.Codice

Blocco(Codice, Anno, Peso, CodiceCava)

- Foreign key: Blocco.CodiceCava → Cava.Codice

Lastra(Numero, Blocco, Superficie, AbitazioneUso\*)

- Foreign key: Lastra.Blocco → Blocco.Codice
- Foreign key: Lastra.Abitazione → Abitazione.Codice

<sup>1</sup> 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!)

Nome: \_\_\_\_\_ Num. Matric.: \_\_\_\_\_

### Esercizio 3: Algebra Relazionale & SQL (7 punti)

Si consideri la seguente base di dati per la registrazione dei concorsi, i candidati e gli esiti:

- CANDIDATO(CF, Nome, Cognome)
- PARTECIPA(CF, CodConcorso, Esito)
- CONCORSO(CodConcorso, Descrizione, Anno)

dove Esito può essere 'positivo' o 'negativo' (usare queste due costanti).

- A. Nel riquadro, scrivere una Query in Algebra Relazione che restituisce i nomi e cognomi di tutti i candidati con esito negativo per almeno un concorso del 2019 (2 punti).<sup>2</sup>

$\Pi_{\text{Nome, Cognome}}(\text{CANDIDATO} \bowtie$   
 $(\sigma_{\text{Esito}='negativo' \text{ AND } \text{Anno}=2019}(\text{CONCORSO} \bowtie \text{PARTECIPA})))$

- B. Nel riquadro, scrivere un'interrogazione SQL che restituisca tutti i codici dei concorsi che hanno avuto solo esiti negativi, senza duplicati (2 punti).

SELECT CodConcorso FROM CONCORSO  
EXCEPT  
SELECT CodConcorso FROM PARTECIPA WHERE  
Esito='Positivo'

oppure

SELECT CodConcorso FROM CONCORSO  
WHERE CodConcorso NOT IN  
(SELECT CodConcorso FROM PARTECIPA  
WHERE Esito='Positivo')

---

<sup>2</sup> 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.

C. Nel Riquadro, scrivere un'interrogazione SQL che restituisca il codice fiscale dei candidati che hanno partecipato a TUTTI i concorsi con descrizione "Banca di Italia" a partire dal 2000 (3 punti)

La seguente è una soluzione ma non l'unica possibile:

```
SELECT CF
FROM PARTECIPA P, CONCORSO C
WHERE P.CodConcorso = C CodConcorso AND
Descrizione='Banca di Italia' AND Anno>2000
GROUP BY CF
HAVING COUNT(*) =

    (SELECT COUNT(*) FROM CONCORSO
     WHERE Descrizione='Banca di Italia'
     AND Anno>2000)
```

#### **Esercizio 4: Transazioni (5 punti)**

Indicare e motivare se lo schedule è conflict-serializzabile

$r_4(y)w_1(z)r_2(y)w_3(x)w_1(y)r_1(x)r_3(z)w_5(z)w_5(y)w_4(z)r_4(x)$

In caso sia conflict-serializzabile, indicare come le transazioni possono essere riordinate per ottenere uno schedule seriale conflict-equivalente.

Il grafo dei conflitti ha un ciclo dalla transazione 4 alla transazione 1, dovuta a  $r_4(y)w_1(z)w_1(y)w_4(z)$ .

Quindi, lo schedule non è conflict-serializzabile.

Nome: \_\_\_\_\_ Num. Matric.: \_\_\_\_\_

### Esercizio 5: Quiz (5 punti)

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

#### Domanda 1 (1.5 Punti)

Data la relazione  $R(\underline{A}, B, C)$  la query  $\text{SELECT } * \text{ FROM } R \text{ ORDER BY } C$ , quale dei seguenti indici velocizza l'esecuzione della query?

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

#### Domanda 2 (1.5 Punti)

Date due relazioni  $R(\underline{A}, B, C)$  e  $S(\underline{D}, E, F)$  dove (1) le uniche chiavi di R e S sono quelle primarie, (2) C è chiave esterna a D e (3) C non può essere NULL. Indicato con  $|X|$  il numero di tuple di una relazione X, il join  $R \bowtie_{C=D} S$  restituisce il seguente numero di tuple:

- (1)  $|R|$
- (2)  $|S|$
- (3)  $|R| * |S|$
- (4) Il minimo tra  $|R|$  e  $|S|$

#### Domanda 3 (2 Punti)

Data la relazione  $R(\underline{A}, B, C)$  con chiave primaria A e chiave addizionale C, date le dipendenze funzionali  $A \rightarrow BC$  e  $C \rightarrow A$ , la relazione è

- (1) In forma normale di Boyce-Codd (BCNF) e in terza forma normale (3NF)
- (2) In 3NF ma non in BCNF
- (3) Né in 3NF, né in BCNF
- (4) In BCNF e ma non in 3NF