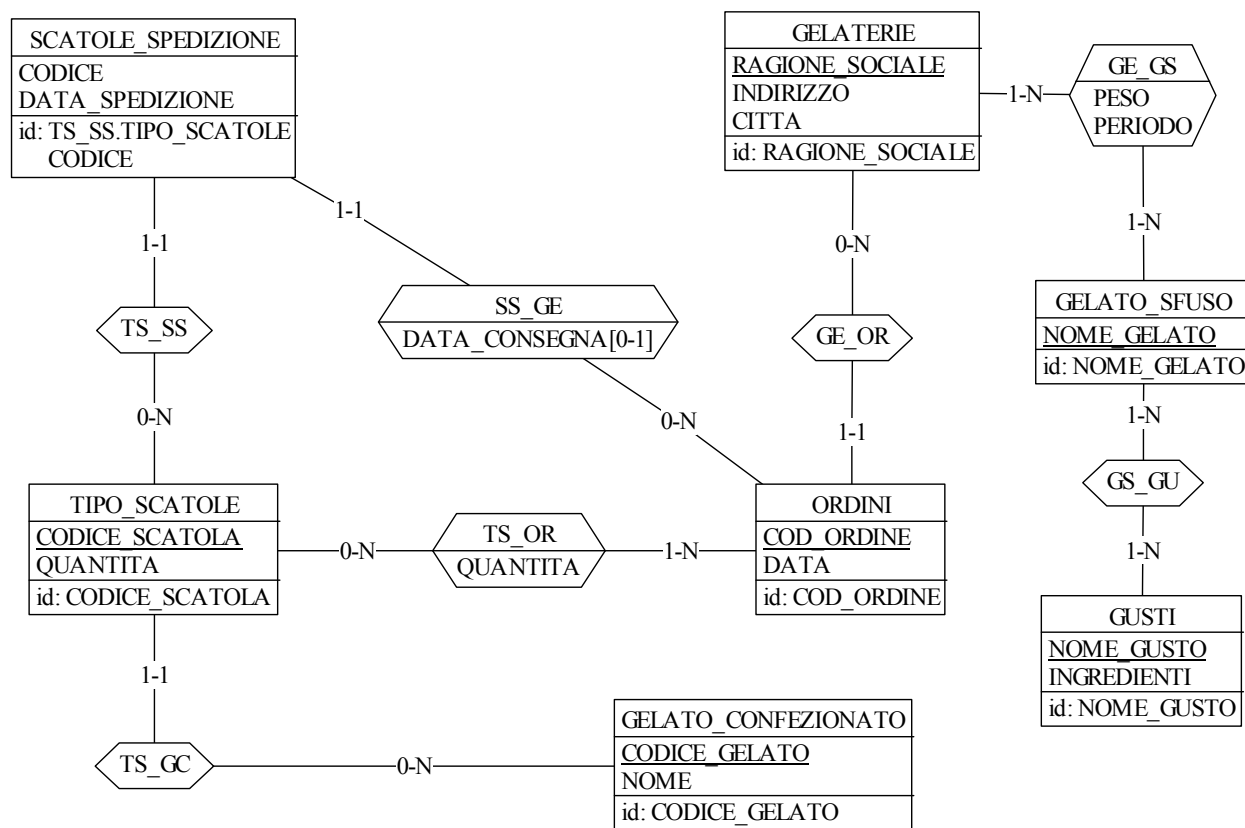


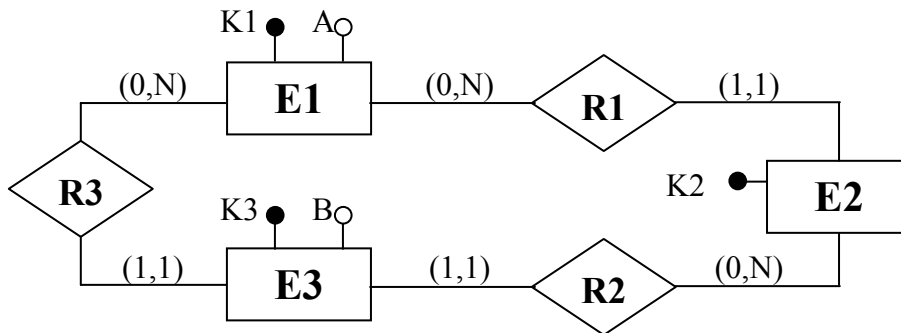
Tempo a disposizione: 2 ore**1) Progettazione concettuale (5 punti)****Commenti:**

- L'entità TIPO_SCATOLE consente di rappresentare ogni tipo di scatola presente nel catalogo; l'entità SCATOLE_SPEDIZIONE conterrà invece una istanza per ogni effettiva scatola spedita.
- L'associazione SS_GE collega SCATOLE_SPEDIZIONE a ORDINI, anziché direttamente a GELATERIE: in questo modo si ha maggiore informazione, ad esempio si può sapere quando un ordine è stato completamente evaso.
- Non è possibile rappresentare il vincolo che le scatole spedite a seguito di un ordine siano dei tipi effettivamente richiesti dall'ordine.

Sistemi Informativi L-B - Prova integrata
15 Aprile 2005
Risoluzione

2) Progettazione logica e normalizzazione (3 punti)

Dato lo schema concettuale in figura



e considerando che:

- tutti gli attributi sono di tipo VARCHAR(255);
- E1, E2 ed E3 vengono tradotte separatamente;
- istanze di E3 sono associate tramite E2 ad istanze di E1 diverse da quelle a cui sono associate tramite R3;

si progettino gli opportuni schemi relazionali e si definiscano **nel database STUDENTI** tali schemi facendo uso dell'SQL di DB2; per gli eventuali vincoli non esprimibili a livello di schema si predispongano opportune **query di verifica da eseguire prima di effettuare inserimenti di tuple**, allo scopo di evitare che tali inserimenti violino i vincoli stessi.

```
CREATE TABLE E1 (  
    K1 VARCHAR(255) NOT NULL PRIMARY KEY,  
    A VARCHAR(255) NOT NULL);
```

```
CREATE TABLE E2 (  
    K2 VARCHAR(255) NOT NULL PRIMARY KEY,  
    KR1 VARCHAR(255) NOT NULL REFERENCES E1);
```

```
CREATE TABLE E3 (  
    K3 VARCHAR(255) NOT NULL PRIMARY KEY,  
    B VARCHAR(255) NOT NULL,  
    KR2 VARCHAR(255) NOT NULL REFERENCES E2,  
    KR3 VARCHAR(255) NOT NULL REFERENCES E1);
```

Query che, data la tupla ('k3', 'b', 'kr2', 'kr3') che si intende inserire in E3, verifica che l'istanza di E1 referenziata tramite E2 sia diversa da quella referenziata tramite R3:

```
SELECT *                                -- OK se non restituisce tuple  
FROM E2  
WHERE E2.K2='kr2'                       -- selezione istanze di E2 referenziate da 'kr2'  
AND E2.KR1='kr3';
```

3) B⁺-tree (2 punti)

Il B⁺-tree, variante del B-tree, è una struttura comunemente usata nei DBMS per realizzare indici paginati, dinamici e multi-livello. Le sue caratteristiche principali sono le seguenti:

- le coppie (valore di chiave, puntatore) sono tutte contenute in pagine (o "nodi") foglia e le foglie sono collegate a lista mediante puntatori per favorire la risoluzione di query di intervallo
- ogni foglia contiene un numero di coppie che varia tra O e 2 * O, dove O è detto "ordine" del B⁺-tree
- ogni percorso dal nodo radice a una foglia ha lunghezza h (altezza del B⁺-tree)
- ogni nodo intermedio (né radice né foglia) ha almeno O + 1 puntatori ad altrettanti nodi figli; la radice, se non è una foglia, ha almeno 2 figli
- un nodo con F + 1 figli contiene F valori di chiave, detti anche "separatori"
- ogni nodo ha al più 2 * O + 1 nodi figli

Per un esempio si rimanda alle dispense del corso.