

# Esame di Basi di Dati

## A.A. 2022/2023 – Appello del 15/02/2023

### Problema 1

Si richiede di progettare lo schema concettuale Entità-Relazione di un'applicazione relativa ad una rete sociale. Degli utenti della rete sociale interessa il codice (identificativo), lo pseudonimo, il livello di socialità e le amicizie con gli altri utenti, ognuna delle quali è identificata dall'utente proponente e dall'utente accettante, con l'ovvia regola che se un utente  $u_1$  è proponente di un'amicizia con l'utente  $u_2$ , quest'ultimo non può essere proponente di un'amicizia con  $u_1$ . Di ogni amicizia l'applicazione aggiorna periodicamente, ma non più di una volta al mese, le relative informazioni, avendo stabilito che il primo aggiornamento corrisponde alla nascita dell'amicizia stessa. Ogni aggiornamento prevede di registrare il numero (che può essere 0) di contatti che i due utenti hanno avuto tra di loro a partire dal precedente aggiornamento e di classificare, a scopo di marketing congiunto, la situazione dell'amicizia secondo due classi: lasca (quando l'amicizia nasce, questa è la classe in cui essa viene posta) e stretta. Quando la situazione di un'amicizia viene classificata come lasca, viene registrato il numero di volte in cui l'amicizia è stata classificata in tale modo e quando viene classificata come stretta, viene registrato quanti "like" dall'ultimo aggiornamento il proponente ha assegnato complessivamente agli altri utenti e quanti ne ha assegnati l'accettante. Di ogni utente "gold" interessa anche l'età ed il compenso annuale che riceve dalla rete sociale. Vale la regola che ogni volta che la situazione di un'amicizia proposta da un utente "gold" viene classificata come stretta, la rete sociale decide il valore di un bonus da assegnare all'utente stesso, valore che è di interesse registrare nella base di dati.

### Problema 2

Si richiede di effettuare la progettazione logica per l'applicazione citata nel problema 1, tenendo conto delle seguenti indicazioni: (i) quando si accede ad un utente si vuole spesso sapere se è un utente "gold" oppure no ed in caso positivo si vuole sempre sapere l'età ed il compenso annuale dell'utente "gold"; (ii) alle informazioni relative a condizioni di amicizie classificate come lasche si accede spesso separatamente rispetto a quelle classificate come strette.

### Problema 3

Riferendosi allo schema logico prodotto per il problema 2, scrivere una query SQL che per ogni coppia  $\langle u_1, u_2 \rangle$  di utenti che forma un'amicizia  $\alpha$  con  $u_1$  proponente e  $u_2$  accettante, restituisca il codice di  $u_1$ , il codice di  $u_2$  ed il carattere "l" o "s" a seconda che nell'ultimo aggiornamento della condizione dell'amicizia  $\alpha$  essa sia stata classificata rispettivamente come lasca o stretta.

### Problema 4

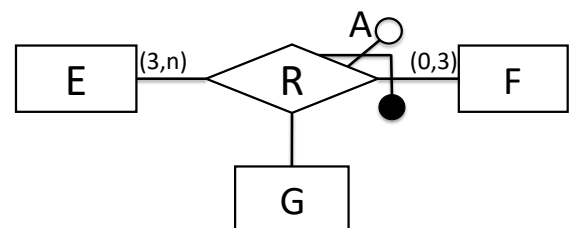
Data la relazione  $R$  il cui schema è  $R(A, B, C)$ , vogliamo verificare se essa soddisfa o no il vincolo di chiave primaria sull'attributo  $A$ . Si chiede di (i) fornire la definizione della nozione "la relazione  $R$  soddisfa il vincolo di chiave primaria sull'attributo  $A$ "; (ii) scrivere una query in algebra relazionale che, data la relazione  $R$  (ovviamente completa delle sue tuple) effettui la verifica suddetta, restituendo l'insieme vuoto in caso di esito positivo della verifica, oppure l'insieme delle tuple che violano il vincolo di chiave primaria in caso di esito negativo.

### Problema 5

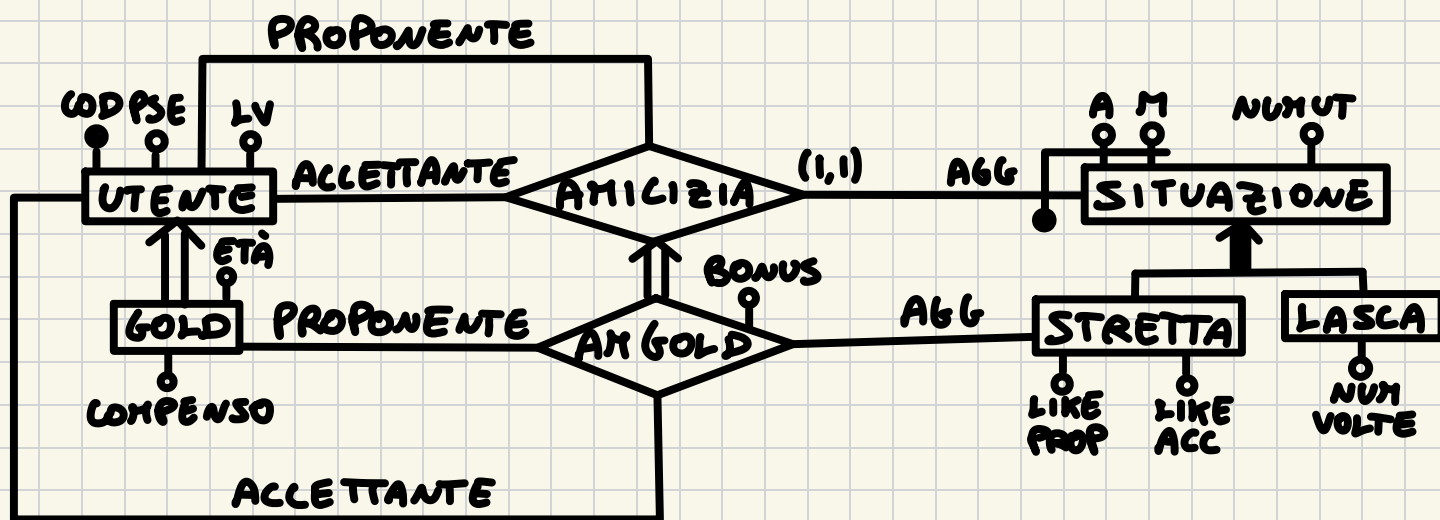
Per lo schema concettuale  $S$  mostrato qui a destra (dove il dominio di  $A$  è  $\{1, 2, 3\}$ ) si consideri una potenziale istanza  $I$  in cui  $istanze(I, E) = \{e_1, e_2\}$  e  $istanze(I, F) = \{f_1, f_2\}$ . Si analizzino poi i seguenti tre casi alternativi:

- (1)  $istanze(I, G) = \emptyset$ ,
- (2)  $istanze(I, G) = \{g_1\}$ ,
- (3)  $istanze(I, G) = \{g_1, g_2\}$

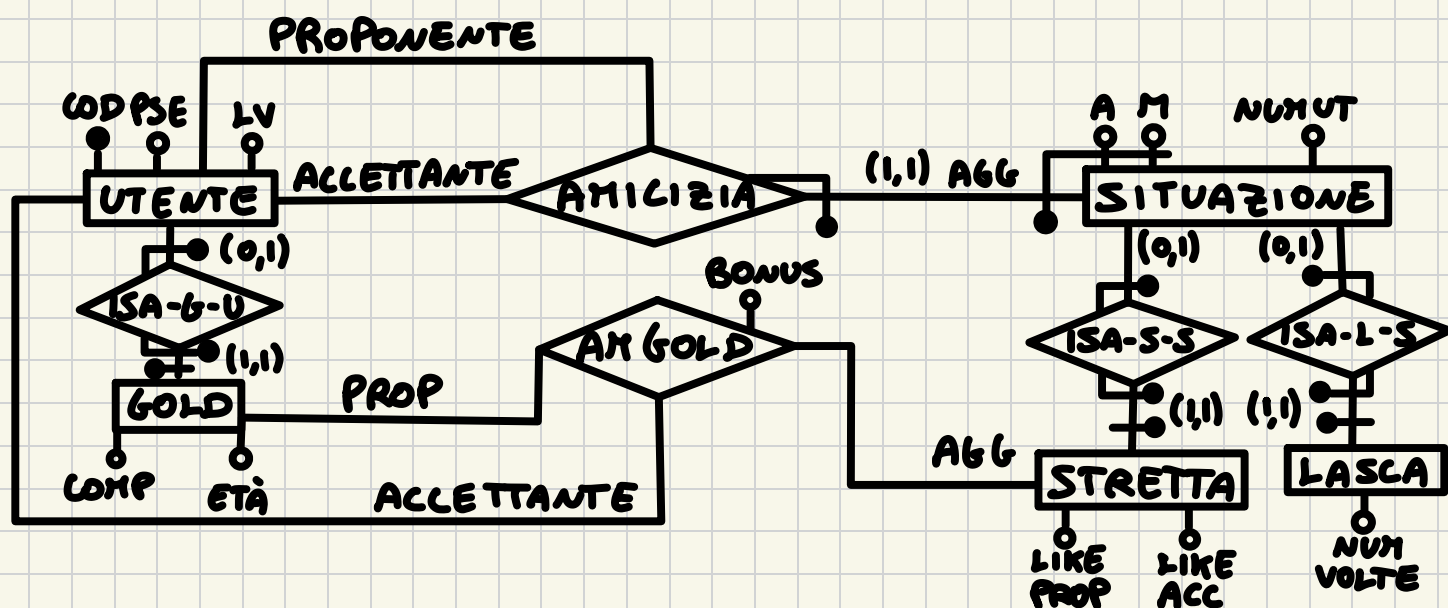
e per ognuno di essi si valuti se è possibile che  $I$  sia effettivamente una istanze corretta per  $S$ , fornendo opportune motivazioni se la risposta è negativa, oppure illustrando i possibili insiemi  $istanze(I, R)$  e  $istanze(I, A)$  che rendono  $I$  una corretta istanza per  $S$  se la risposta è positiva.



# PROBLEMA 1



# PROBLEMA 2



UTENZE (COD, PSEUDONIMO, LIVELLO)

SITUAZIONE (PROP, ACC, ANNO, MESE, NUM UTENTI)

FK: SIT[PROP]  $\subseteq$  PROP[COD]

FK: SIT[ACC]  $\subseteq$  ACC[COD]

VINC: SIT[P, A, A, M] = L[P, A, A, M]  $\cup$  S[P, A, A, M]

LASCA (PROP, ACC, ANNO, MESE, NUM VOLTE)

FK: L[PROP, ACC, ANNO, MESE]  $\subseteq$  SIT[PROP, ACC, ANNO, MESE]

VINC: L[P, ACC, AN, M]  $\cap$  STR[P, ACC, AN, M] =  $\emptyset$

STRETTA (PROP, ACC, ANNO, MESE, LIKE PROP, LIKE ACC)

FK: S[P, A, A, M]  $\subseteq$  SIT[P, A, A, M]

GOLD (UTENTE, COMPENSO, ETÀ)

FK: G[UT]  $\subseteq$  UT[COD]

AMICIZIA GOLD (PROP, ACC, MESE, ANNO, BONUS)

FK: AG[PROP]  $\subseteq$  GOLD[UT]

FK: AG[ACC]  $\subseteq$  UT[COD]

FK: AG[PROP, ACC, MESE, ANNO]  $\subseteq$  STRETTA[P, A, M, A]



i ALLORPIAMO UTENTE E GOLD

UTENZE(COD, PSEUDONIMO, LIVELLO, COMPENSO, ETÀ)

VINCOLO DI TUPLA: ETÀ IS NULL  $\Leftrightarrow$  COMPENSO IS NULL

ii SITUAZIONE LASCA (PROP, ACC, ANNO, MESE, NUM UTENTI)

FK: SIT[PROP]  $\subseteq$  PROP[COD]

FK: SIT[ACC]  $\subseteq$  ACC[COD]

VINL. SIT[P, A, A, M] = L[P, A, A, M]  $\cup$  S[P, A, A, M]

SITUAZIONE STRETTA (PROP, ACC, ANNO, MESE, NUM UTENTI)

FK: SIT[PROP]  $\subseteq$  PROP[COD]

FK: SIT[ACC]  $\subseteq$  ACC[COD]

VINL. SIT[P, A, A, M] = L[P, A, A, M]  $\cup$  S[P, A, A, M]

LASCA (PROP, ACC, ANNO, MESE, NUM VOLTE)

FK: L[PROP, ACC, ANNO, MESE]  $\subseteq$  SIT LASCA[PROP, ACC, ANNO, MESE]

VINC: L[P, ACC, AN, M]  $\cap$  STR[P, ACC, AN, M] =  $\emptyset$

STRETTA (PROP, ACC, ANNO, MESE, LIKE PROP, LIKE ACC)

FK: S[P, A, A, M]  $\subseteq$  SIT STRETTA[P, A, A, M]

ALLORPIAMO

LASCA (PROP, ACC, ANNO, MESE, NUM UTENTI, NUM VOLTE)

FK: LA [PROP]  $\subseteq$  UT [COD]

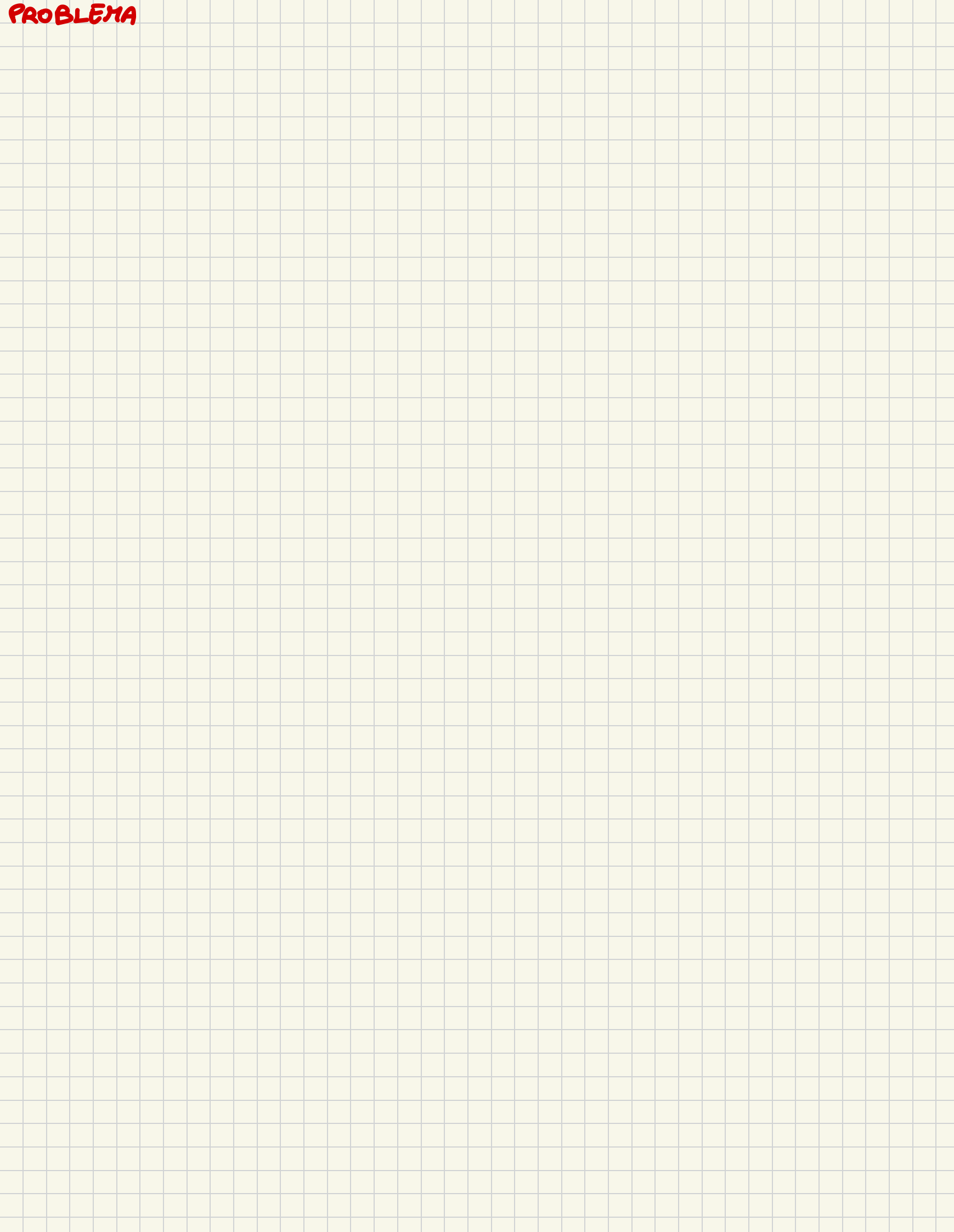
FK: LA [ACC]  $\subseteq$  ACC [COD]

STRETTA (PROP, ACC, ANNO, MESE, NUM UTENTI, LIKEP, LIKEACC)

FK: ST [PROP]  $\subseteq$  UT [COD]

FK: ST [ACC]  $\subseteq$  UT [COD]

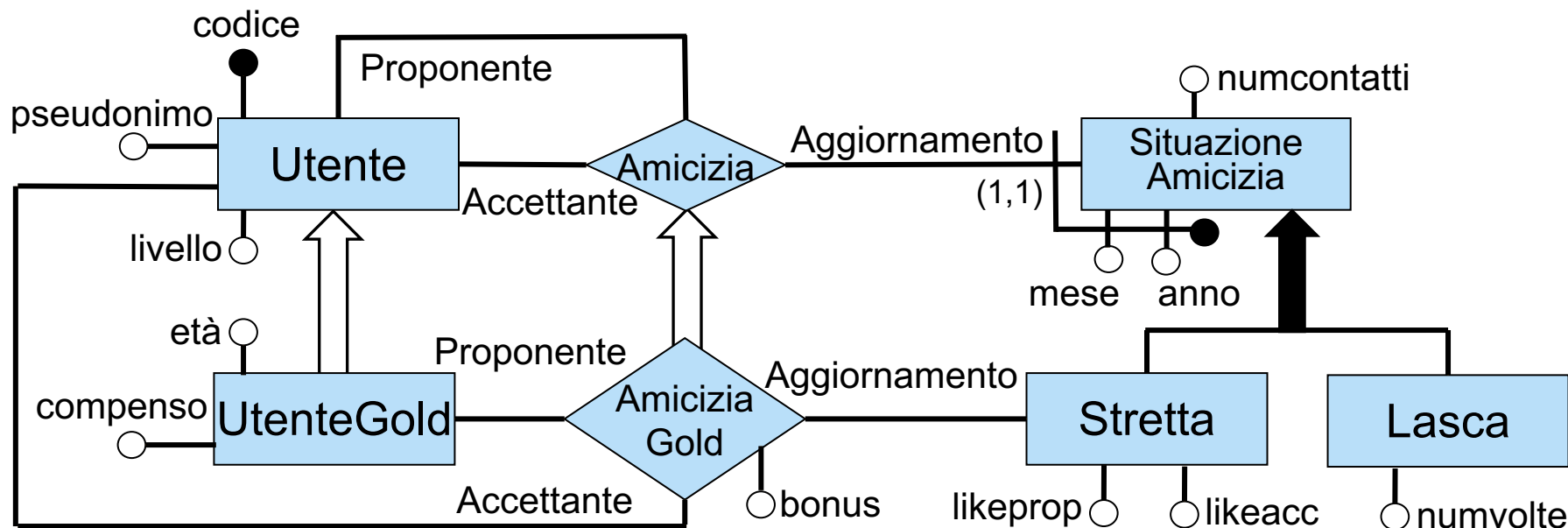
# PROBLEMA



## Nota al problema 1

La mancanza dei vincoli esterni che presenteremo nello schema concettuale e nello schema logico corrispondenti alla soluzione del problema 1 non comporta perdita di punti in sede di valutazione del compito d'esame.

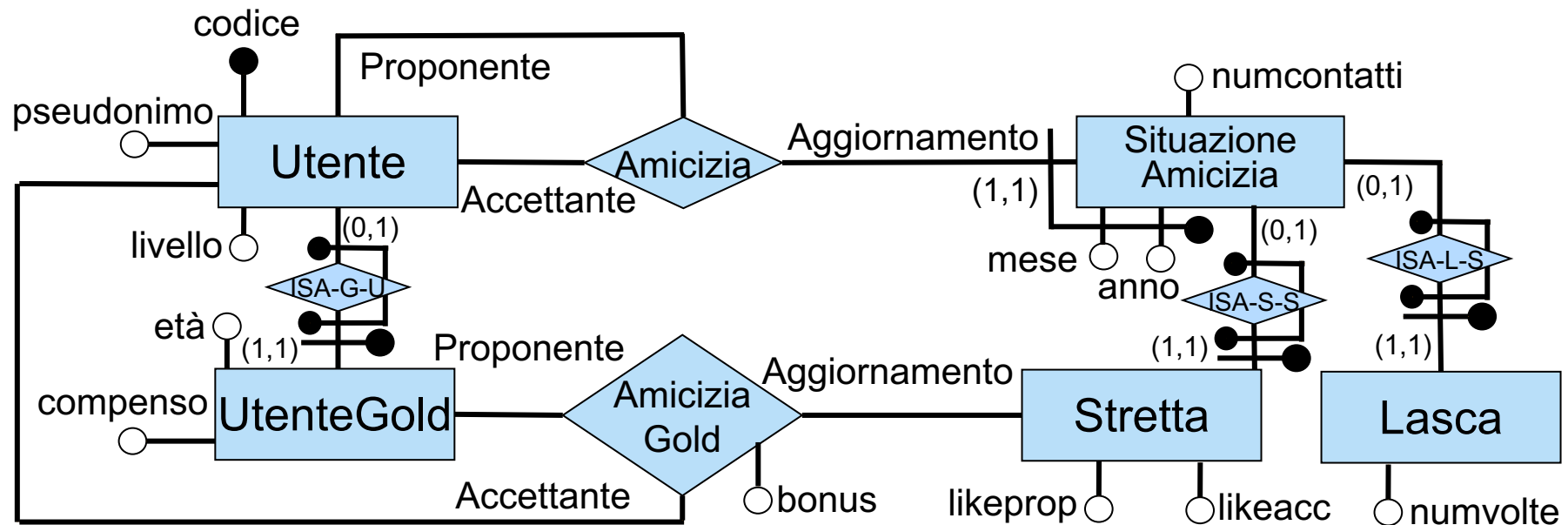
# Problema 1 – Schema concettuale



## Vincoli esterni:

- 1) se  $\langle \text{Proponente}:x, \text{Accettante}:y, \text{Aggiornamento}:z \rangle$  è istanza di **Amicizia**, allora non esiste alcuna  $w$  tale che  $\langle \text{Proponente}:y, \text{Accettante}:x, \text{Aggiornamento}:w \rangle$  lo è;
- 2) se  $\langle \text{Proponente}:x, \text{Accettante}:y, \text{Aggiornamento}:z \rangle$  è istanza di **Amicizia** e  $z$  ha  $\langle \text{anno}, \text{mese} \rangle$  minimo rispetto alle istanze di **SituazioneAmicizia** legate tramite **Amicizia** a **Proponente}:x** e **Accettante}:y**, allora  $z$  è istanza di **Lasca**;
- 3) per ogni istanza  $z$  di **Stretta**, se l'istanza  $\langle \text{Proponente}:x, \text{Accettante}:y, \text{Aggiornamento}:z \rangle$  di **Amicizia** è tale che  $x$  è istanza di **UtenteGold**, allora  $\langle \text{Proponente}:x, \text{Accettante}:y, \text{Aggiornamento}:z \rangle$  è anche istanza di **AmiciziaGold**.

## Problema 2 – Schema concettuale ristrutturato



### Vincoli esterni:

- 1) se  $\langle \text{Proponente}:x, \text{Accettante}:y, \text{Aggiornamento}:z \rangle$  è istanza di **Amicizia**, allora non esiste alcuna  $w$  tale che  $\langle \text{Proponente}:y, \text{Accettante}:x, \text{Aggiornamento}:w \rangle$  lo è;
- 2) se  $\langle \text{Proponente}:x, \text{Accettante}:y, \text{Aggiornamento}:z \rangle$  è istanza di **Amicizia** e  $z$  ha  $\langle \text{anno}, \text{mese} \rangle$  minimo rispetto alle istanze di **SituazioneAmicizia** legate tramite **Amicizia** a **Proponente}:x** e **Accettante}:y**, allora  $z$  partecipa ad **ISA-L-C**;
- 3) per ogni istanza  $\langle \text{Proponente}:x, \text{Accettante}:y, \text{Aggiornamento}:z \rangle$  di **AmiciziaGold** tale che  $\langle \text{Stretta}:z, \text{SituazioneAmicizia}:w \rangle$  è istanza di **ISA-S-S**, si ha che istanza  $\langle \text{Proponente}:x, \text{Accettante}:y, \text{Aggiornamento}:w \rangle$  è istanza di **Amicizia**;
- 4) per ogni istanza  $z$  di **Stretta** tale che  $\langle \text{Stretta}:z, \text{SituazioneAmicizia}:w \rangle$  è istanza di **ISA-S-S**, se l'istanza  $\langle \text{Proponente}:x, \text{Accettante}:y, \text{Aggiornamento}:w \rangle$  di **Amicizia** è tale che  $x$  è istanza di **UtenteGold**, allora  $\langle \text{Proponente}:x, \text{Accettante}:y, \text{Aggiornamento}:z \rangle$  è anche istanza di **AmiciziaGold**;
- 5) **vincolo di generalizzazione**: ogni istanza di **SituazioneAmicizia** partecipa esattamente ad una istanza tra **ISA-S-S** e **ISA-L-S**.

## Problema 2 – Traduzione diretta

Schema logico prodotto  
dalla traduzione diretta:

Utente(codice,livello,pseudonimo)

SituazioneAmicizia(proponente,accettante,mese,anno,numcontatti)

**foreign key:** SituazioneAmicizia[proponente]  $\subseteq$  Utente[codice]

**foreign key:** SituazioneAmicizia[accettante]  $\subseteq$  Utente[codice]

**vincolo di generalizzazione:** SituazioneAmicizia[proponente,accettante,mese,anno] =  
Stretta[proponente,accettante,mese,anno]  $\cup$  Lasca[proponente,accettante,mese,anno]

**vincolo intra-relazionale:** se  $t$  è in SituazioneAmicizia, allora non esiste  $t'$  in SituazioneAmicizia tale che  
 $t$ [proponente] =  $t$ [accettante] e  $t$ [accettante] =  $t$ [proponente]

**vincolo inter-relazionale:** se  $t$  è in SituazioneAmicizia e non esiste  $t'$  in SituazioneAmicizia con  $t$ '[proponente] =  
 $t$ [proponente],  $t$ '[accettante] =  $t$ [accettante] e  $\langle t$ [anno,mese] $\rangle$  minore di  $\langle t'$ [anno,mese] $\rangle$ ,  
allora  $t$  è in Lasca[proponente,accettante,mese,anno]

Lasca(proponente,accettante,mese,anno,numvolte)

**foreign key:** Lasca[proponente,accettante,mese,anno]  $\subseteq$  SituazioneAmicizia[proponente,accettante,mese,anno]

**vincolo di disgiunzione:** Lasca[proponente,accettante,mese,anno]  $\cap$  Stretta[proponente,accettante,mese,anno] =  $\emptyset$

Stretta(proponente,accettante,mese,anno,likeprop,likeacc)

**foreign key:** Stretta[proponente,accettante,mese,anno]  $\subseteq$  SituazioneAmicizia[proponente,accettante,mese,anno]

**vincolo inter-relazionale:** (select proponente,accettante,mese,anno  
from Stretta  
where proponente in (select codice from UtenteGold)  $\subseteq$   
AmiciziaGold[proponente,accettante,mese,anno])

UtenteGold(codice,età,compenso)

**foreign key:** UtenteGold[codice]  $\subseteq$  Utente[codice]

AmiciziaGold(proponente,accettante,mese,anno,bonus)

**foreign key:** AmiciziaGold[proponente]  $\subseteq$  UtenteGold[codice]

**foreign key:** AmiciziaGold[proponente,accettante,mese,anno]  $\subseteq$  Stretta[proponente,accettante,mese,anno]



# Problema 2 – Ristrutturazione dello schema logico

## Schema logico prodotto dalla ristrutturazione:

1. La prima indicazione di progetto induce un accorpamento tra Utente ed UtenteGold.
2. La seconda indicazione induce una decomposizione di SituazioneAmicizia in due fasce corrispondenti a Stretta e Lasca e poi un accorpamento tra la fascia Lasca e la relazione Lasca ed un accorpamento tra la fascia Stretta e la relazione Stretta.

Utente(codice,livello,pseudonimo,età\*,compenso\*)

**vincolo di tupla:** età is null se e solo se compenso is null

Lasca(proponente,accettante,mese,anno,numcontatti,numvolte)

**foreign key:** Lasca[proponente]  $\subseteq$  Utente[codice]

**foreign key:** Lasca[accettante]  $\subseteq$  Utente[codice]

**vincolo inter-relazionale:** se t è una tupla in Lasca o Stretta, allora nessun t' esiste in Stretta o Lasca con  
t'[proponente] = t[accettante] e t'[accettante] = t[proponente]

**vincolo inter-relazionale:** se t è una tupla in Lasca tale che nessun t' esiste in Lasca con  
t'[proponente] = t[proponente], t'[accettante] = t[accettante] e t'[anno,mese] <= t[anno,mese],  
allora nessun t'' esiste in Stretta con t''[proponente] = t[proponente], t''[accettante] = t[accettante]  
e t''[anno,mese] <= t[anno,mese]

Stretta(proponente,accettante,mese,anno,numcontatti,likeprop,likeacc)

**foreign key:** Stretta[proponente]  $\subseteq$  Utente[codice]

**foreign key:** Stretta[accettante]  $\subseteq$  Utente[codice]

**vincolo di disgiunzione:** Stretta[proponente,accettante,mese,anno]  $\cap$  Lasca[proponente,accettante,mese,anno] =  $\emptyset$

**vincolo inter-relazionale:** (select proponente,accettante,mese,anno from Stretta  
where proponente in (select codice from UtenteGold)  $\subseteq$   
AmiciziaGold[proponente,accettante,mese,anno])

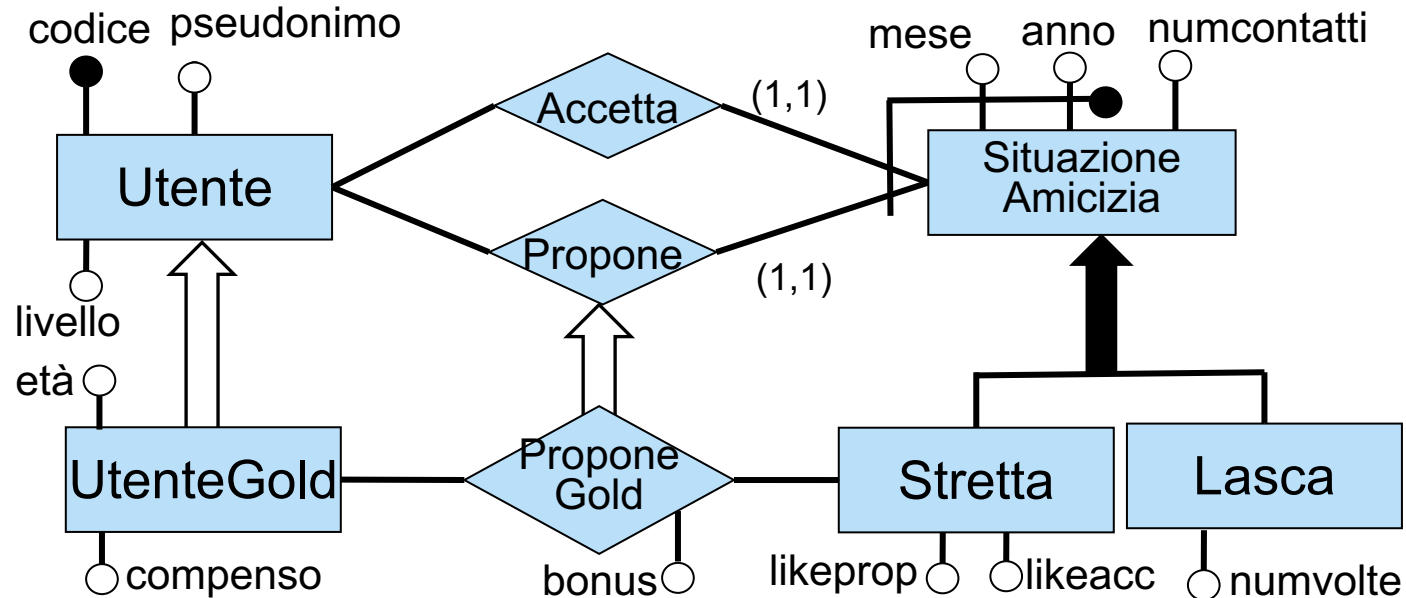
AmiciziaGold(proponente,accettante,mese,anno,bonus)

**vincolo inter-relazionale:** AmiciziaGold[proponente]  $\subseteq$  (select codice from Utente where età is not null)

**foreign key:** AmiciziaGold[proponente,accettante,mese,anno]  $\subseteq$  Stretta[proponente,accettante,mese,anno]

# Problema 1 – Schema concettuale alternativo 1

NOTA: mostriamo qui sotto uno schema concettuale alternativo, di fatto equivalente a quello presentato prima e quindi perfettamente corretto come soluzione

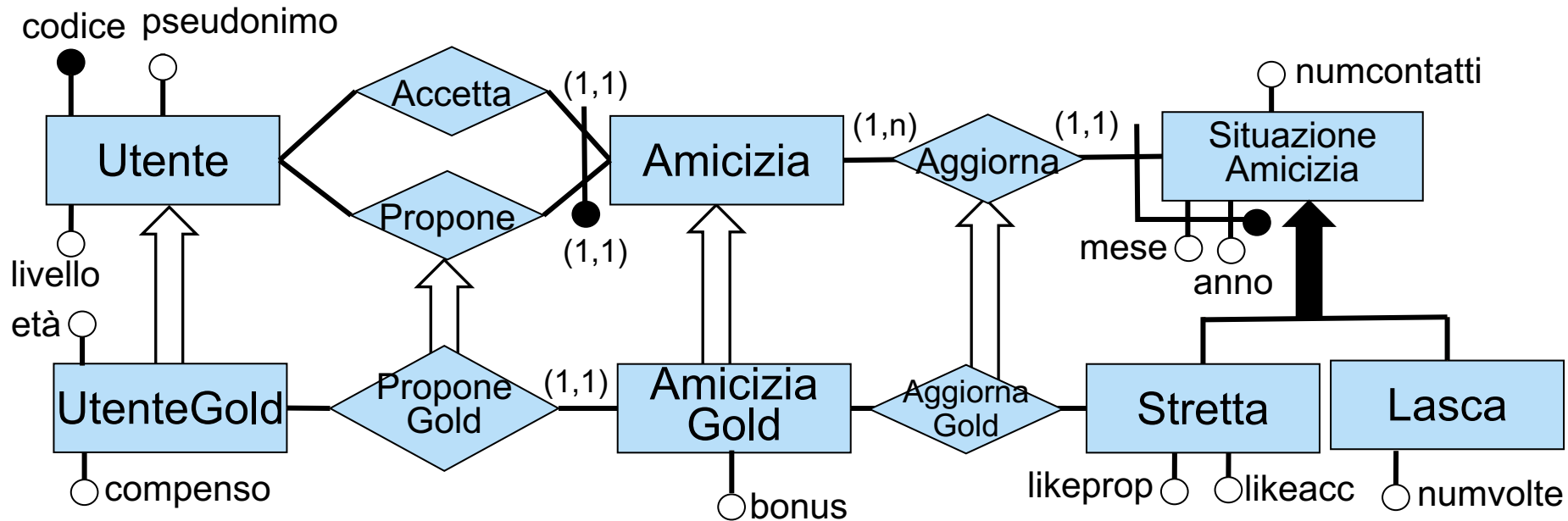


## Vincoli esterni:

- 1) se  $\langle \text{Utente}:x, \text{SituazioneAmicizia}:y \rangle$  è istanza di **Accetta** e  $\langle \text{Utente}:z, \text{SituazioneAmicizia}:y \rangle$  è istanza di **Propone**, allora non esiste alcuna  $w$  tale che  $\langle \text{Utente}:z, \text{SituazioneAmicizia}:w \rangle$  è istanza di **Accetta** e  $\langle \text{Utente}:x, \text{SituazioneAmicizia}:w \rangle$  è istanza di **Propone**;
- 2) se  $y$  è istanza di **SituazioneAmicizia** e  $y$  ha  $\langle \text{anno}, \text{mese} \rangle$  minimo rispetto alle istanze di **SituazioneAmicizia** legate tramite **Accetta** e **Propone** agli stessi utenti, allora  $y$  è istanza di **Lasca**;
- 3) per ogni istanza  $z$  di **Stretta**, se  $\langle \text{Utente}:y, z:\text{SituazioneAmicizia} \rangle$  è istanza di **Propone** e  $y$  è istanza di **UtenteGold**, allora  $\langle \text{UtenteGold}:y, z:\text{Stretta} \rangle$  è istanza di **ProponeGold**.

# Problema 1 – Schema concettuale alternativo 2

NOTA: mostriamo qui sotto un secondo schema concettuale alternativo, di fatto equivalente a quello presentato prima e quindi perfettamente corretto come soluzione



## Vincoli esterni:

- 1) se  $\langle \text{Utente}:x, \text{Amicizia}:y \rangle$  è istanza di **Accetta** e  $\langle \text{Utente}:z, \text{Amicizia}:y \rangle$  è istanza di **Propone**, allora non esiste alcuna  $w$  tale che  $\langle \text{Utente}:z, \text{Amicizia}:w \rangle$  è istanza di **Accetta** e  $\langle \text{Utente}:x, \text{Amicizia}:w \rangle$  è istanza di **Propone**;
- 2) se  $\langle \text{Amicizia}:x, \text{SituazioneAmicizia}:y \rangle$  è istanza di **Aggiorna** e  $y$  ha  $\langle \text{anno}, \text{mese} \rangle$  minimo rispetto alle istanze di **SituazioneAmicizia** legate tramite **Aggiorna** a  $x$ , allora  $y$  è istanza di **Lasca**;
- 3) per ogni istanza  $z$  di **Stretta**, se  $\langle \text{Amicizia}:y, z:\text{SituazioneAmicizia} \rangle$  è istanza di **Aggiorna**,  $\langle \text{Utente}:x, \text{Amicizia}:y \rangle$  è istanza di **Propone** e  $x$  è istanza di **UtenteGold**, allora  $\langle \text{Amicizia}:y, z:\text{Stretta} \rangle$  è istanza di **AmiciziaGold** e  $\langle \text{Utente}:x, \text{AmiciziaGold}:y \rangle$  è istanza di **ProponeGold**.

## Problema 3 – testo e soluzione

**Testo:** Riferendosi allo schema logico prodotto per il problema 2, scrivere una query SQL che per ogni coppia  $\langle u1, u2 \rangle$  di utenti che forma un'amicizia  $\alpha$  con  $u1$  proponente e  $u2$  accettante, restituisca il codice di  $u1$ , il codice di  $u2$  ed il carattere "l" o "s" a seconda che nell'ultimo aggiornamento della Situazione dell'amicizia  $\alpha$  essa sia stata classificata rispettivamente come lasca o stretta.

## Soluzione:

[illegible]

## Problema 4 – soluzione

**Testo:** Data la relazione R il cui schema è  $R(A,B,C)$ , vogliamo verificare se essa soddisfa o no il vincolo di chiave primaria sull'attributo A. Si chiede di (i) fornire la definizione della nozione "la relazione R soddisfa il vincolo di chiave primaria sull'attributo A"; (ii) scrivere una query in algebra relazionale che, data la relazione R (ovviamente completa delle sue tuple) effettui la verifica suddetta, restituendo l'insieme vuoto in caso di esito positivo della verifica, oppure l'insieme delle tuple che violano il vincolo di chiave primaria in caso di esito negativo.

### Soluzione:

(i) La relazione R soddisfa il vincolo di chiave primaria sull'attributo A se in A non compaiono valori nulli e se non esistono due tuple  $t_1$  e  $t_2$  in R tali che  $t_1$  e  $t_2$  sono diverse e  $t_1[A] = t_2[A]$ , ossia tali che  $t_1[A] = t_2[A]$  e  $t_1[B] \neq t_2[B]$  oppure  $t_1[C] \neq t_2[C]$ .

(ii) La query in algebra relazionale deve semplicemente verificare la condizione espressa nella definizione riportata sopra, restituendo quelle tuple, se esistono, che violano la condizione stessa. Le tuple che violano la condizione dei valori nulli si trovano banalmente con una selezione. Le tuple che hanno in R una diversa tupla che coincide in A si trovano con un join di R con sé stessa (opportunamente ridenominata) usando la condizione di uguaglianza su A e disuguaglianza su B o C:

$SEL_{A \text{ is null}}(R)$   
 $\cup$

$PROJ_{A,B,C} (R \text{ JOIN}_{A=A1 \text{ and } (B \neq B1 \text{ or } C \neq C1)} REN_{A1 \leftarrow A, B1 \leftarrow B, C1 \leftarrow C}(R))$

## Problema 5 – soluzione

Analizziamo separatamente i tre casi.

(1)  $\text{istanze}(I, G) = \emptyset$

Chiaramente, in questo caso l'istanza  $I$  di  $S$  non è corretta, perché  $\text{istanze}(I, G) = \emptyset$  implica  $\text{istanze}(I, R) = \emptyset$ , condizione che è però incompatibile con la partecipazione obbligatoria ad  $R$  delle due istanze  $e_1$  ed  $e_2$  di  $E$ .

(2)  $\text{istanze}(I, G) = \{g_1\}$

Anche in questo caso l'istanza  $I$  di  $S$  non è corretta. Consideriamo infatti l'istanza  $e_1$  in  $\text{istanze}(I, E)$  e proviamo a formare le 3 tuple etichettate (sono 3 per il vincolo di cardinalità su  $E$ ) in  $\text{istanze}(I, R)$  che hanno  $e_1$  nel ruolo  $E$ :

- mettiamo la prima tupla  $\langle E:e_1, G:g_1, F:f_1 \rangle$  in  $\text{istanze}(I, R)$ ,
- mettiamo la seconda tupla  $\langle E:e_1, G:g_1, F:f_2 \rangle$  in  $\text{istanze}(I, R)$

e a questo punto non possiamo più formare la terza tupla, visto che non abbiamo più istanze di  $G$  o  $F$  per formare una tupla diversa da quelle che già sono in  $\text{istanze}(I, R)$ . Ricordiamo, infatti, che  $\text{istanze}(I, R)$ , indipendentemente dal vincolo di identificazione su  $R$  che coinvolge l'attributo  $A$ , deve essere un insieme e quindi non può avere duplicati. Concludiamo che l'istanza  $I$  di  $S$  non è corretta nel caso 2.

(3)  $\text{istanze}(I, G) = \{g_1, g_2\}$

In questo caso è facile verificare che gli insiemi:

$\text{istanze}(I, R) = \{ \langle E:e_1, G:g_1, F:f_1 \rangle, \langle E:e_1, G:g_1, F:f_2 \rangle, \langle E:e_1, G:g_2, F:f_1 \rangle, \langle E:e_2, G:g_1, F:f_1 \rangle, \langle E:e_2, G:g_1, F:f_2 \rangle, \langle E:e_2, G:g_2, F:f_2 \rangle \}$

$\text{istanze}(I, A) = \{ \langle \langle E:e_1, G:g_1, F:f_1 \rangle, 1 \rangle, \langle \langle E:e_1, G:g_1, F:f_2 \rangle, 1 \rangle, \langle \langle E:e_1, G:g_2, F:f_1 \rangle, 2 \rangle, \langle \langle E:e_2, G:g_1, F:f_1 \rangle, 3 \rangle, \langle \langle E:e_2, G:g_1, F:f_2 \rangle, 2 \rangle, \langle \langle E:e_2, G:g_2, F:f_2 \rangle, 3 \rangle \}$

rendono  $I$  una istanza corretta di  $S$ . In particolare,  $\text{istanze}(I, R)$  è un insieme (non ha duplicati) e sia i vincoli di cardinalità sia il vincolo di identificazione sono soddisfatti dall'istanza  $I$ .