Name

CWID

Midterm Exam

March 7th, 2022
10:00-11:15

CS520 - Data Integration, Warehousing, and Provenance

Results

Please leave this empty!

1.1		1.2		1.3		Sum	

Instructions

- Try to answer all the questions using what you have learned in class. Keep hard questions until the end.
- **When writing a query, write the query in a way that it would work over all possible database instances and not just for the given example instance!**
- The exam is closed book and closed notes! No calculator, smartphones, or similar allowed!

Consider the following database schema and example instance about students, their interests, organizations, and events organized by organizations:

student

sid	firstname	lastname	major	gpa
1	alice	smith	cs	3.5
2	alice	muller	psych	4.0
3	bob	smith	cs	2.5
4	malice	peters	phys	3.5

interest

student	activity
1	surfing
1	hacking
2	chess
2	tennis
3	surfing
3	hacking

organization

name	nummembers
acm	40
iit tennis club	30
the hacking surfers	3

event

name	organization	activity	date
chihack	acm	hacking	2022-03-02
potluck	acm	food	2022-01-11
iit annual surfing context	the hacking surfers	surfing	2012-03-15
surfhackaton	the hacking surfers	hacking	2022-04-05

Hints:

- Attributes with black background form the primary key of a relation (e.g., *sid* for relation *student*)
- The attributes *student* of relation *interest* are a foreign key to relation *student*.
- The attributes *organization* of relation *event* are a foreign key to relation *organization*.

Part 1.1 Datalog (Total: 38 Points)

Recall that Datalog applies set semantics.

Question 1.1.1 (5 Points)

Write a **Datalog program** that returns the *lastname* and *gpa* of students that study *cs*.

Solution

```
q(L,G) :- student(_,_,L,cs,G).
```

Question 1.1.2 (6 Points)

Write a **Datalog program** that returns the *firstname* and *lastname* of students which are interested in *surfing* or *hacking*.

Solution

```
q(F,L) :- student(S,F,L,_,_), interest(S,surfing).
q(F,L) :- student(S,F,L,_,_), interest(S,hacking).
```

Question 1.1.3 (9 Points)

Write a **Datalog program** that returns the *sid* of students which are only interested in *hacking*, i.e., they have *hacking* as an interest, but do not have any other interests.

Solution

```
hackers(S) :- interest(S,hacking).
otherinterests(S) :- interest(S,I), I != hacking.
q(S) :- hackers(S), not otherinterests(S).
```

Question 1.1.4 (9 Points)

Write a **Datalog program** that returns pairs of students (*sid* and *lastname*) that may be interested in the same organization (*name*). A student is considered to be interested in an organization, if the organization hosts at least one event with an activity that matches one of the students interest. For instance, organization *acm* has *hacking* events which both *Alice Smith* and *Malice Peters* would be interested in.

Solution

```
studentorg(S,L,O) :- student(S,_,L,_,_), interest(S,A), event(_,O,A,_).
q(S1,L1,S2,L2) :- studentorg(S1,L1,O), studentorg(S2,L2,O).
```

Question 1.1.5 (9 Points)

Write a **Datalog program** that returns pairs of students (their *sids*) which have exactly the same interests. For example, in the given example database instance, *alice smith* and *malice peters* have the same interests (*hacking* and *surfing*).

Solution

```
sids(S) :- student(S,_,_,_,_).
differentinterest(S1,S2) :- interest(S1,I), sids(S2), not interest(S2,I).
differentinterest(S1,S2) :- interest(S2,I), sids(S1), not interest(S1,I).
studentpairs(S1,S2) :- sids(S1), sids(S2).
q(S1,S2) :- studentpairs(S1,S2), not differentinterest(S1,S2).
```

Part 1.2 Constraints (Total: 26 Points)

Question 1.2.1 Expressing Constraints in First-Order Logic (13 Points)

Recall the representation of constraints as universally quantified first-order logic implications as introduced in class. Write down the logical encoding of the following constraints over the example schema:

- The foreign key from attribute student of relation interest to relation student.
- No organization can have more than 50 members.
- All food events (with *activity* equal to *food*) of organizations with more than 10 members have to take place after 2022-05-01.
- The following functional dependency for relation event: $organization, date \to activity$

Solution

$$FK_1 : \forall s, a : interest(s, i) \to \exists x_1, x_2, x_3, x_4 : student(s, x_1, x_2, x_3, x_4)$$

$$LESS50 : \forall o, m : organization(o, m) \to m \leq 50$$

$$FOODLATE : \forall n, o, a, d, m : event(n, o, a, d) \wedge n = food \wedge organization(o, m) \wedge m > 10 \to d > 2022 - 05 - 01$$

$$FD : \forall n_1, o, a_1, d, n_2, a_2 : event(n_1, o, a_1, d) \wedge event(n_2, o, a_2, d) \to a_1 = a_2$$

Question 1.2.2 Creating Denial Constraints (13 Points)

Create denial constraints over the example schema based on the following descriptions.

- No organization can have more then 50 members.
- The organization ACM can only organize events with *hacking* or *food* activities
- No student can have more than 2 interests.

Solution

$$d_1 : \forall n, m \neg (organization(n, m) \land m > 50)$$

$$d_2 : \forall n, o, a, d : \neg (event(n, o, a, d) \land o = acm \land a \neq hacking \land a \neq food)$$

$$d_3 \forall s, a_1, a_2, a_3 : \neg (interest(s, a_1) \land interest(s, a_2) \land interest(s, a_3) \land a_1 \neq a_2 \land a_2 \neq a_3 \land a_1 \neq a_3)$$

Part 1.3 Query Containment And Equivalence (Total: 36 Points)

Question 1.3.1 (36 Points)

Consider the queries shown below. Check all possible containment relationships. If there exists a containment mapping from Q_i to Q_j then write down the mapping.

Q_1(Z,A) :- R(X,Z), S(Z,A), R(Z,Y).
Q_2(Y,Y) :- S(Y,Y), R(Z,A), R(Y,B), R(A,Y).
Q_3(X,Y) :- S(X,Y), R(A,X), R(X,Z).
Q_4(A,B) :- R(A,C), R(C,A), S(A,B).

Solution

$Q_1 \rightarrow Q_2$:

$Y \rightarrow B$
$X \rightarrow A$
$Z \rightarrow Y$
$A \rightarrow Y$

$Q_1 \rightarrow Q_3$:

$Y \rightarrow Z$
$X \rightarrow A$
$Z \rightarrow X$
$A \rightarrow Y$

$Q_1 \rightarrow Q_4$:

$Y \rightarrow C$
$X \rightarrow C$
$Z \rightarrow A$
$A \rightarrow B$

$Q_2 \rightarrow Q_1$:
no containment mapping exists

$Q_2 \rightarrow Q_3$:
no containment mapping exists

$Q_2 \rightarrow Q_4$:
no containment mapping exists

$Q_3 \rightarrow Q_1$:

$Y \rightarrow A$
$X \rightarrow Z$
$Z \rightarrow Y$
$A \rightarrow X$

$Q_3 \rightarrow Q_2$:

$Y \rightarrow Y$
$X \rightarrow Y$
$Z \rightarrow B$
$A \rightarrow A$

$Q_3 \rightarrow Q_4$:

$Y \rightarrow B$
$X \rightarrow A$
$Z \rightarrow C$
$A \rightarrow C$

$Q_4 \rightarrow Q_1$:
no containment mapping exists

$Q_4 \rightarrow Q_2$:
no containment mapping exists

$Q_4 \rightarrow Q_3$:
no containment mapping exists