1. Satisfiability, Safety, and Containment

Instructions: Work in groups of 2 students. You can write up your answers by hand (provided your handwriting is legible) or use a word processing system like Latex or Word. Note that experience has shown that Word is in general difficult to use for this kind of task. If you prefer to write up your solution by hand, submit a scanned electronic version. Please, include name and email address in your submission.

1. Finite vs. Infinite Satisfiability

We consider first-order sentences (= closed formulas), possibly with constants and equality, but without function symbols. A sentence is *finitely satisfiable* if it is satisfied by some interpretation with a finite domain. It is *infinitely satisfiable* if it is satisfied by some interpretation with an infinite domain.
For each of the two cases below, write down a sentence ϕ, ψ, respectively, with the required property:

1. ϕ is finitely satisfiable, but not infinitely satisfiable;
2. ψ is infinitely satisfiable, but not finitely satisfiable.

Explain why your formulas have the required property.

(10 Points)

2. Finite vs. Database Satisfiability

We consider now first-order sentences that may have constants, but no function symbols and no equality or disequality atoms. We say that a sentence is *database satisfiable* if it is satisfied by an interpretation that is a database instance. (Note that database and finite satisfiability are not necessarily the same since every database instance has the domain **dom**, which is infinite.)
Do there exist sentences ϕ, ψ such that

1. ϕ is finitely satisfiable, but not database satisfiable;
2. ψ is database satisfiable, but not finitely satisfiable?

For each of the two cases, if there exists such a formula, write one down. If there does not exist such a formula, write down a proof for your claim. (That is, explain why a sentence is finitely satisfiable if it is database satisfiable, or why it is database satisfiable if it is finitely satisfiable.)

Hint: One case is easy, while the other one is complicated.

(16 Points)

3. Positive Queries

We consider formulas without built-in predicates such as "$=$, $\leq$, or $\neq$."
A first-order logic formula is *positive* if it contains only the logical symbols "$\wedge$", "$\vee$", and "$\exists$". A relational calculus query Q_ϕ is *positive* if the defining formula ϕ is positive.

1. Is satisfiability of positive queries decidable? If yes, what does an algorithm look like? If not, how can one prove undecidability?

   (8 Points)

2. Are positive queries safe?

   (6 Points)

3. Can one represent positive queries in relational algebra? If one can, explain how. If not, provide a proof.

   (4 Points)

4. Query Semantics and Integrity Constraints

Let Σ be the signature with the schemas

$$\mathsf{S}(\mathsf{theater}, \mathsf{mtitle}), \qquad \mathsf{M}(\mathsf{title}, \mathsf{director})$$

Intuitively S stands for "schedule" and M stands for "movie". Both attributes, title and mtitle, refer to the title of a movie.
We want to look at two attempts to formulate the intuitive query:

> "Return the theaters that play only movies by Tarantino."

Consider the following two first-order formulas with free variable t:

$$\phi_1 = \exists m\, \mathsf{S}(t,m) \land \forall m' \,(\mathsf{S}(t,m') \rightarrow \mathsf{M}(m', \text{'Tarantino'}))$$

$$\phi_2 = \exists m\, \mathsf{S}(t,m) \land \forall m', d\,(\mathsf{S}(t,m') \land \mathsf{M}(m', d) \rightarrow d = \text{'Tarantino'})).$$

and the corresponding two relational calculus queries

$$Q_1 = \{\, t \mid \phi_1 \,\}$$

$$Q_2 = \{\, t \mid \phi_2 \,\}.$$

We want to find out whether whether the two queries are equivalent.

1. Is one of the two queries Q_1, Q_2 contained in the other?

   If you claim that $Q_i \sqsubseteq Q_j$, provide an argument (not necessarily a formal proof). If you claim that $Q_i \not\sqsubseteq Q_j$, give a database instance $\mathbf{I}$ such that $Q_i(\mathbf{I}) \not\subseteq Q_j(\mathbf{I})$.

(6 Points)

Consider in addition the following two integrity constraints:

$$\gamma_K = \forall\, m, d, d' \,(\mathsf{M}(m,d) \land \mathsf{M}(m,d') \rightarrow d = d')$$

$$\gamma_{FK} = \forall\, t, m\, \exists\, d\, (\mathsf{S}(t,m)) \rightarrow \mathsf{M}(m,d)).$$

Clearly, γ_K is a primary key constraint that states that title is the primary key of M, while γ_{FK} is a foreign key constraint that states that the mtitle attribute of S refers to the key attribute title of M.

2. What can you say about the containment of the two queries if you consider only instances that satisfy one or both of the constraints γ_K and γ_{FK}?

   Note that these are three cases. For each case, two possible containments have to be considered.

(10 Points)

Submission: Thursday, 5 April 2018, 1 pm, by email to

`Werner DOT Nutt AT unibz DOT it.`