Querying Inconsistent Description Logic Knowledge Bases under Preferred Repair Semantics

Meghyn Bienvenu and **Camille Bourgaux**
LRI, CNRS & Université Paris-Sud
Orsay, France

François Goasdoué
IRISA, Université de Rennes 1
Lannion, France

Abstract

Recently several inconsistency-tolerant semantics have been introduced for querying inconsistent description logic knowledge bases. Most of these semantics rely on the notion of a repair, defined as an inclusion-maximal subset of the facts (ABox) which is consistent with the ontology (TBox). In this paper, we study variants of two popular inconsistency-tolerant semantics obtained by replacing classical repairs by various types of preferred repair. We analyze the complexity of query answering under the resulting semantics, focusing on the lightweight logic DL-Lite$_{\mathcal{R}}$. Unsurprisingly, query answering is intractable in all cases, but we nonetheless identify one notion of preferred repair, based upon priority levels, whose data complexity is "only" coNP-complete. This leads us to propose an approach combining incomplete tractable methods with calls to a SAT solver. An experimental evaluation of the approach shows good scalability on realistic cases.

1 Introduction

Description logic (DL) knowledge bases consist of an ontology (called a TBox) expressing conceptual knowledge about a particular domain and a dataset (or ABox) containing facts about particular individuals (Baader et al. 2003). Recent years have seen an increasing interest in performing database-style query answering over DL knowledge bases. Since scalability is crucial in data-rich applications, there has been a trend to using so-called lightweight DLs for which query answering is tractable w.r.t. the size of the ABox. Particular attention has been paid to DLs of the DL-Lite family (Calvanese et al. 2007) which possess the notable property that query answering can be reduced to evaluation of standard database queries.

An important issue that arises in the context of DL query answering is how to handle the case in which the ABox is inconsistent with the TBox. Indeed, while it may be reasonable to assume that the TBox has been properly debugged, the ABox will typically be very large and subject to frequent modifications, both of which make errors likely. Since it may be too difficult or costly to identify and fix these errors, it is essential to be able to provide meaningful answers to queries in the presence of such data inconsistencies. Unfortunately, standard DL semantics is next to useless in such circumstances, as everything is entailed from an inconsistent knowledge base. To address this issue, several different inconsistency-tolerant semantics have been proposed for querying inconsistent DL knowledge bases. Among them, the AR and IAR semantics (Lembo et al. 2010) are the most well-known and well-studied. Both semantics are based upon the notion of a *repair*, defined as an inclusion-maximal subset of the ABox which are consistent with the TBox. The AR semantics amounts to computing those answers that hold no matter which repair is chosen, whereas the more cautious IAR semantics queries the intersection of the repairs.

When additional information on the reliability of ABox assertions is available, it is natural to use this information to identify *preferred repairs*, and to use the latter as the basis of inconsistency-tolerant query answering. In this paper, we investigate variants of the AR and IAR semantics obtained by replacing the classical notion of repair by one of four different types of preferred repairs. Cardinality-maximal repairs are intended for settings in which all ABox assertions are believed to have the same likelihood of being correct. The other three types of preferred repairs target the scenario in which some assertions are considered to be more reliable than others, which can be captured qualitatively by partitioning the ABox into *priority levels* (and then applying either the set inclusion or cardinality criterion to each level), or quantitatively by assigning *weights* to the ABox assertions (and selecting those repairs having the greatest weight).

The first contribution of the paper is a systematic study of the complexity of answering conjunctive and atomic queries under the eight resulting preferred repair semantics. We focus on the lightweight logic DL-Lite$_{\mathcal{R}}$ that underlies the OWL 2 QL profile (Motik et al. 2012), though many of our results can be generalized to other data-tractable ontology languages. For the IAR semantics, the use of preferred repairs significantly impacts complexity: we move from polynomial data complexity in the case of (plain) IAR semantics to coNP-hard data complexity (or worse) for IAR semantics based on preferred repairs. For the AR semantics, query answering is known to be coNP-complete in data complexity already for the standard notion of repairs, but adding preferences often results in even higher complexities. The sole exception is $\subseteq_P$-repairs (which combine priority levels and the set inclusion criterion), for which both AR and IAR query answering are "only" coNP-complete in data complexity.

Our second contribution is a practical approach to query answering in DL-Lite$_{\mathcal{R}}$ under the AR, $\subseteq_P$-AR, and $\subseteq_P$-IAR semantics. We first show how to encode query answering under these three semantics in terms of propositional unsatisfiability, using a reachability analysis to reduce the size of the encodings. In the CQAPri system we have implemented, a subset of the query results is computed using incomplete tractable methods, and a SAT solver is used to determine the status of the remaining possible answers. An experimental evaluation demonstrates the scalability of the approach in settings we presume realistic. This positive empirical result is due in large part to the efficacy of the incomplete methods, which leave only few cases to be handled by the SAT solver.

Proofs and further details on the experiments can be found in the appendix.

2 Preliminaries

We briefly recall the syntax and semantics of description logics (DLs), and some relevant notions from complexity.

Syntax A DL *knowledge base (KB)* consists of an ABox and a TBox, both of which are constructed from a set $\mathsf{N_C}$ of *concept names* (unary predicates), a set of $\mathsf{N_R}$ of *role names* (binary predicates), and a set $\mathsf{N_I}$ of *individuals* (constants). The *ABox* (dataset) consists of a finite number of *concept assertions* of the form $A(a)$ with $A \in \mathsf{N_C}, a \in \mathsf{N_I}$ and *role assertions* of the form $R(a, b)$ with $R \in \mathsf{N_R}, a, b \in \mathsf{N_I}$. The *TBox* (ontology) consists of a set of axioms whose form depends on the DL in question.

In DL-Lite$_{\mathcal{R}}$, TBox axioms are either *concept inclusions* $B \sqsubseteq C$ or *role inclusions* $Q \sqsubseteq S$ formed using the following syntax (where $A \in \mathsf{N_C}$ and $R \in \mathsf{N_R}$):

$$B := A \mid \exists Q,\ C := B \mid \neg B,\ Q := R \mid R^-,\ S := Q \mid \neg Q$$

Semantics An *interpretation* has the form $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty set and $\cdot^{\mathcal{I}}$ maps each $a \in \mathsf{N_I}$ to $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, each $A \in \mathsf{N_C}$ to $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, and each $R \in \mathsf{N_R}$ to $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The function $\cdot^{\mathcal{I}}$ is straightforwardly extended to general concepts and roles, e.g. $(R^-)^{\mathcal{I}} = \{(c, d) \mid (d, c) \in R^{\mathcal{I}}\}$ and $(\exists Q)^{\mathcal{I}} = \{c \mid \exists d : (c, d) \in Q^{\mathcal{I}}\}$. An interpretation $\mathcal{I}$ satisfies an inclusion $G \sqsubseteq H$ if $G^{\mathcal{I}} \subseteq H^{\mathcal{I}}$; it satisfies $A(a)$ (resp. $R(a, b)$) if $a^{\mathcal{I}} \in A^{\mathcal{I}}$ (resp. $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$). An interpretation $\mathcal{I}$ is a *model* of $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ if $\mathcal{I}$ satisfies all inclusions in $\mathcal{T}$ and assertions in $\mathcal{A}$. A KB $\mathcal{K}$ is *consistent* if it has a model, and we say that an ABox $\mathcal{A}$ is $\mathcal{T}$*-consistent* if the KB $\langle \mathcal{T}, \mathcal{A} \rangle$ is consistent.

Queries Our main focus will be on *conjunctive queries* (CQs) which take the form $\exists \vec{y}\, \psi$, where ψ is a conjunction of atoms of the forms $A(t)$ or $R(t, t')$, where t, t' are variables or individuals, and $\vec{y}$ is a tuple of variables from ψ. A CQ is called *Boolean* if all of its variables are existentially quantified; a CQ consisting of a single atom is an *atomic query* (AQ). When we use the generic term *query*, we mean a CQ. A Boolean CQ q is *entailed* from $\mathcal{K}$, written $\mathcal{K} \models q$, just in the case that q holds in all models of $\mathcal{K}$. For a non-Boolean CQ q with free variables $x_1, \ldots, x_k$, a tuple of individuals $\mathbf{a} = (a_1, \ldots, a_k)$ is a *(certain) answer* for q w.r.t. $\mathcal{K}$ just in the case that $\mathcal{K} \models q[\mathbf{a}]$, where $q[\mathbf{a}]$ is the Boolean query obtained by replacing each x_i by a_i. Thus, CQ answering is straightforwardly reduced to entailment of Boolean CQs. For this reason, we can focus w.l.o.g. on the latter problem.

Complexity There are two common ways of measuring the complexity of query entailment: *combined complexity* is with respect to the size of the whole input $(\mathcal{T}, \mathcal{A}, q)$, whereas *data complexity* is only with respect to the size of $\mathcal{A}$.

In addition to the well-known complexity classes P, NP, and coNP, our results will involve the following classes in the polynomial hierarchy: Δ^p_2 (polynomial time using an NP oracle), $\Delta^p_2[O(\log n)]$ (polynomial time with at most logarithmically many NP oracle calls), Σ^p_2 (non-deterministic polynomial time with an NP oracle) and its complement Π^p_2.

The following result resumes known results on the complexity of reasoning in DL-Lite$_{\mathcal{R}}$ under classical semantics.

Theorem 1 (Calvanese et al. 2007)**.** *In DL-Lite$_{\mathcal{R}}$, consistency and AQ entailment are in* P *w.r.t. combined complexity, and CQ entailment is in* P *w.r.t. data complexity and co*NP*-complete w.r.t. combined complexity.*

3 Preferred Repair Semantics

In this section, we recall two important inconsistency-tolerant semantics and introduce variants of these semantics based upon different notions of preferred repairs. For simplicity, we state the definitions in terms of query entailment.

A central notion in inconsistency-tolerant query answering is that of a repair, which corresponds to a minimal way of modifying the ABox so as to restore consistency. Typically, minimality is defined in terms of set inclusion, yielding:

Definition 1. A *repair* of a KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ is an inclusion-maximal subset of $\mathcal{A}$ that is $\mathcal{T}$-consistent.

Several inconsistency-tolerant semantics have been proposed based on this notion of repair. The most well-known, and arguably the most natural, is the AR semantics (Lembo et al. 2010), which was inspired by consistent query answering in relational databases (cf. (Bertossi 2011) for a survey).

Definition 2. A Boolean query q is entailed by $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ under *AR semantics* if $\langle \mathcal{T}, \mathcal{B} \rangle \models q$ for every repair $\mathcal{B}$ of $\mathcal{K}$.

The intuition underlying the AR semantics is as follows. In the absence of further information, we cannot identify the "correct" repair, and so we only consider a query to be entailed if it can be obtained from each of the repairs.

The IAR semantics (Lembo et al. 2010) adopts an even more cautious approach: only assertions which belong to every repair (or equivalently, are not involved in any contradiction) are considered when answering the query.

Definition 3. A Boolean query q is entailed by a KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ under *IAR semantics* if $\langle \mathcal{T}, \mathcal{B}_\cap \rangle \models q$ where $\mathcal{B}_\cap$ is the intersection of all repairs of $\mathcal{K}$.

It is easy to see that every query that is entailed under IAR semantics is also entailed under AR semantics, but the converse does not hold in general.

The above notion of repair integrates a very simple preference relation, namely set inclusion. When additional information on the reliability of ABox assertions is available, it is natural to use this information to identify *preferred repairs*, and to use the latter as the basis of inconsistency-tolerant

	$\subseteq$	$\leq$	$\subseteq_P$	$\leq_P$	$\leq_w$
AR	coNP	$\mathbf{\Delta_2^p[O(\log n)]}$	**coNP**	$\mathbf{\Delta_2^p}^\dagger$	$\mathbf{\Delta_2^p}^\dagger$
IAR	in P	$\mathbf{\Delta_2^p[O(\log n)]}$	**coNP**	$\mathbf{\Delta_2^p}^\dagger$	$\mathbf{\Delta_2^p}^\dagger$

(a) Data complexity

	$\subseteq$	$\leq$	$\subseteq_P$	$\leq_P$	$\leq_w$
AR	Π_2^p	$\mathbf{\Pi_2^p}$	$\mathbf{\Pi_2^p}$	$\mathbf{\Pi_2^p}$	$\mathbf{\Pi_2^p}$
IAR	NP	$\mathbf{\Delta_2^p[O(\log n)]}$	$\mathbf{\Delta_2^p[O(\log n)]}$	$\mathbf{\Delta_2^p}^\dagger$	$\mathbf{\Delta_2^p}^\dagger$

(b) Combined complexity

Figure 1: Data and combined complexity of CQ entailment over DL-Lite KBs under AR and IAR semantics for different types of preferred repairs. For AQs, the data and combined complexity coincide with the data complexity for CQs. All results are completeness results unless otherwise noted. New results in bold. † $\Delta_2^p[O(\log n)]$-complete under the assumption that there is a bound on the number of priority classes (resp. maximal weight).

reasoning. This idea leads us generalize the earlier definitions[1], using preorders to model preference relations.

Definition 4. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a KB, and let $\preceq$ be a preorder over subsets of $\mathcal{A}$. A *$\preceq$-repair of* $\mathcal{K}$ is a $\mathcal{T}$-consistent subset of $\mathcal{A}$ which is maximal w.r.t. $\preceq$. The set of $\preceq$-repairs of $\mathcal{K}$ is denoted $Rep_\preceq(\mathcal{K})$.

Definition 5. A Boolean query q is entailed by $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ under the *$\preceq$-AR semantics*, written $\mathcal{K} \models_{\preceq\text{-AR}} q$, if $\langle \mathcal{T}, \mathcal{B} \rangle \models q$ for every $\mathcal{B} \in Rep_\preceq(\mathcal{K})$; it is entailed by $\mathcal{K}$ under the *$\preceq$-IAR semantics*, written $\mathcal{K} \models_{\preceq\text{-IAR}} q$, if $\langle \mathcal{T}, \mathcal{B}_\cap \rangle \models q$ where $\mathcal{B}_\cap = \bigcap_{\mathcal{B} \in Rep_\preceq(\mathcal{K})} \mathcal{B}$.

In this paper, we consider four standard ways of defining preferences over subsets, cf. (Eiter and Gottlob 1995).

Cardinality ($\leq$) A first possibility is to compare subsets using set cardinality: $\mathcal{A}_1 \leq \mathcal{A}_2$ iff $|\mathcal{A}_1| \leq |\mathcal{A}_2|$. The resulting notion of $\leq$-repair is appropriate when all assertions are believed to have the same (small) likelihood of being erroneous, in which case repairs with the largest number of assertions are most likely to be correct.

Priority levels ($\subseteq_P$, $\leq_P$) We next consider the case in which ABox assertions have been partitioned into priority levels $\mathcal{P}_1, \ldots, \mathcal{P}_n$ based on their perceived reliability, with assertions in $\mathcal{P}_1$ considered most reliable, and those in $\mathcal{P}_n$ least reliable. Such a prioritization can be used to separate a part of the dataset that has already been validated from more recent additions. Alternatively, one might stratify assertions based upon the concept or role names they use (when some predicates are known to be more reliable), or the data sources from which they originate (in information integration applications). Given a prioritization $P = \langle \mathcal{P}_1, \ldots, \mathcal{P}_n \rangle$ of $\mathcal{A}$, we can refine the $\subseteq$ and $\leq$ preorders as follows:

- Prioritized set inclusion: $\mathcal{A}_1 \subseteq_P \mathcal{A}_2$ iff $\mathcal{A}_1 \cap \mathcal{P}_i = \mathcal{A}_2 \cap \mathcal{P}_i$ for every $1 \leq i \leq n$, or there is some $1 \leq i \leq n$ such that $\mathcal{A}_1 \cap \mathcal{P}_i \subsetneq \mathcal{A}_2 \cap \mathcal{P}_i$ and for all $1 \leq j < i$, $\mathcal{A}_1 \cap \mathcal{P}_j = \mathcal{A}_2 \cap \mathcal{P}_j$.
- Prioritized cardinality: $\mathcal{A}_1 \leq_P \mathcal{A}_2$ iff $|\mathcal{A}_1 \cap \mathcal{P}_i| = |\mathcal{A}_2 \cap \mathcal{P}_i|$ for every $1 \leq i \leq n$, or there is some $1 \leq i \leq n$ such that $|\mathcal{A}_1 \cap \mathcal{P}_i| < |\mathcal{A}_2 \cap \mathcal{P}_i|$ and for all $1 \leq j < i$, $|\mathcal{A}_1 \cap \mathcal{P}_j| = |\mathcal{A}_2 \cap \mathcal{P}_j|$.

Notice that a single assertion on level $\mathcal{P}_i$ is preferred to any number of assertions from $\mathcal{P}_{i+1}$, so these preorders are best suited for cases in which there is a significant difference in the perceived reliability of adjacent priority levels.

Weights ($\leq_w$) The reliability of different assertions can be modelled quantitatively using a function $w : \mathcal{A} \to \mathbb{N}$ assigning weights to the ABox assertions. The weight function w induces a preorder $\leq_w$ over subsets of $\mathcal{A}$ in the expected way: $\mathcal{A}_1 \leq_w \mathcal{A}_2$ iff $\sum_{\alpha \in \mathcal{A}_1} w(\alpha) \leq \sum_{\alpha \in \mathcal{A}_2} w(\alpha)$. If the ABox is populated using information extraction techniques, the weights may be derived from the confidence levels output by the extraction tool. Weight-based preorders can also be used in place of the $\leq_P$ preorder to allow for compensation between the priority levels.

4 Complexity Results

In this section, we study the complexity of query entailment under preferred repair semantics. We focus on knowledge bases formulated in DL-Lite$_\mathcal{R}$, since it is a popular DL for OBDA applications and the basis for OWL 2 QL (Motik et al. 2012). However, many of our results hold also for other DLs and ontology languages (see end of section).

Figure 1 recalls existing results for query entailment under the standard AR and IAR semantics and presents our new results for the different preferred repair semantics.

Theorem 2. *The results in Figure 1 hold.*

Proof idea. The upper bounds for AR-based semantics involve guessing a preferred repair that does not entail the query; for the IAR-based semantics, we guess preferred repairs that omit some ABox assertions and verify that the query is not entailed from the remaining assertions. For the lower bounds, we were able to adapt some proofs from (Bienvenu 2012; Bienvenu and Rosati 2013); the $\Delta_2^p[O(log\ n)]$ lower bounds proved the most challenging and required significant extensions of existing reductions. □

Let us briefly comment on the obtained results. Concerning data complexity, we observe that for preferred repairs, the IAR semantics is just as difficult as the AR semantics. This is due to the fact that there is no simple way of computing the intersection of preferred repairs, whereas this task is straightforward for $\subseteq$-repairs. However, if we consider combined complexity, we see that the IAR semantics still retains some computational advantage over AR semantics. This lower complexity translates into a concrete practical advantage: for the IAR semantics, one can precompute the intersection of repairs in an offline phase and then utilize standard querying algorithms at query time, whereas no such

[1] Ours is not the first work to consider preferred repairs – see Section 7 for references and discussion.

precomputation is possible for the AR semantics. Finally, if we compare the different types of preferred repairs, we find that the $\subseteq_P$ preorder leads to the lowest complexity, and $\leq_P$ and $\leq_w$ the greatest. However, under the reasonable assumption that there is a bound on the number of priority classes (resp. maximal weight), we obtain the same complexity for the semantics based on $\leq$-, $\leq_P$- and $\leq_w$-repairs.

We should point out that the only properties of DL-Lite$_\mathcal{R}$ which are used in the upper bound proofs are those stated in Theorem 1. Consequently, our combined complexity upper bounds apply to all ontology languages having polynomial combined complexity for consistency and instance checking and NP combined complexity for CQ entailment, and in particular to the OWL 2 EL profile (Motik et al. 2012). Our data complexity upper bounds apply to all data-tractable ontology languages, which includes Horn DLs (Hustadt, Motik, and Sattler 2007; Eiter et al. 2008) and several dialects of Datalog +/- (Calì, Gottlob, and Lukasiewicz 2012).

5 Query Answering via Reduction to SAT

In this section, we show how to answer queries over DL-Lite$_\mathcal{R}$ KBs under $\subseteq_P$-AR and $\subseteq_P$-IAR semantics by translation to propositional unsatisfiability. We chose to focus on $\subseteq_P$-repairs as they offer the lowest complexity among the different forms of preferred repairs, and seem natural from the point of view of applications.

To simplify the presentation of our encodings, we introduce the notions of conflicts of a KB and causes of a query.

Definition 6. A *conflict* for $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ is a minimal $\mathcal{T}$-inconsistent subset of $\mathcal{A}$. A *cause* for a Boolean CQ q is a minimal $\mathcal{T}$-consistent subset $\mathcal{C} \subseteq \mathcal{A}$ such that $\langle \mathcal{T}, \mathcal{C} \rangle \models q$.

Fact 1. In DL-Lite$_\mathcal{R}$, conflicts have cardinality at most two.

The encodings presented in this section use variables to represent ABox assertions, so that each valuation corresponds to a subset of the ABox. In the case of the $\subseteq_P$-AR semantics, the most obvious encoding would stipulate that the subset corresponding to a valuation (i) contains no cause for q, (ii) is maximal w.r.t. $\subseteq_P$, and (iii) contains no conflicts. However, such an encoding would contain as many variables as ABox facts, even though most of the ABox may be irrelevant for answering the query at hand.

In order to identify potentially relevant assertions, we introduce the notion of an oriented conflict graph (inspired by the conflict hypergraphs from (Chomicki, Marcinkowski, and Staworko 2004)). In what follows, we use $\alpha \preceq_P \beta$ to signify that there exist $i \leq j$ such that $\alpha \in \mathcal{P}_i$ and $\beta \in \mathcal{P}_j$.

Definition 7. Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a DL-Lite$_\mathcal{R}$ KB and P be a prioritization of $\mathcal{A}$. The *oriented conflict graph for $\mathcal{K}$ and P*, denoted $G^P_\mathcal{K}$, is the directed graph whose set of vertices is $\mathcal{A}$ and which has an edge from β to α whenever $\alpha \preceq_P \beta$ and $\{\alpha, \beta\}$ is a conflict for $\mathcal{K}$.

We now give a more succinct encoding, which can be seen as selecting a set of assertions which contradict all of the query's causes (thereby ensuring that no cause is present), and verifying that this set can be extended to a $\subseteq_P$-repair. Importantly, to check the latter, it suffices to consider only those assertions that are reachable in $G^P_\mathcal{K}$ from an assertion that contradicts some cause.

Theorem 3. *Let q be a Boolean CQ , $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a DL-Lite$_\mathcal{R}$ KB, and $P = \langle \mathcal{P}_1, \ldots, \mathcal{P}_n \rangle$ be a prioritization of $\mathcal{A}$. Consider the following propositional formulas having variables of the form x_α for $\alpha \in \mathcal{A}$:*

$$\varphi_{\neg q} = \bigwedge_{\mathcal{C} \in \mathsf{causes}(q)} \Big(\bigvee_{\alpha \in \mathcal{C}} \bigvee_{\substack{\beta \in \mathsf{confl}(\alpha) \\ \beta \preceq_P \alpha}} x_\beta \Big)$$

$$\varphi_{\mathsf{max}} = \bigwedge_{\alpha \in R_q} \Big(x_\alpha \vee \bigvee_{\substack{\beta \in \mathsf{confl}(\alpha) \\ \beta \preceq_P \alpha}} x_\beta \Big)$$

$$\varphi_{\mathsf{cons}} = \bigwedge_{\substack{\alpha, \beta \in R_q \\ \beta \in \mathsf{confl}(\alpha)}} \neg x_\alpha \vee \neg x_\beta$$

where $\mathsf{causes}(q)$ contains all causes for q in $\mathcal{K}$, $\mathsf{confl}(\alpha)$ contains all assertions β such that $\{\alpha, \beta\}$ is a conflict for $\mathcal{K}$, and R_q is the set of assertions reachable in $G^P_\mathcal{K}$ from some assertion β such that x_β appears in $\varphi_{\neg q}$. Then $\mathcal{K} \models_{\subseteq_P\text{-AR}} q$ iff $\varphi_{\neg q} \wedge \varphi_{\mathsf{max}} \wedge \varphi_{\mathsf{cons}}$ is unsatisfiable.

We observe that for the plain AR semantics, we can further simplify the encoding by dropping the formula φ_{max}.

For the $\subseteq_P$-IAR semantics, a query is not entailed just in the case that every cause is absent from some $\subseteq_P$-repair. This can be tested by using one SAT instance per cause.

Theorem 4. *Let q, $\mathcal{K}$, P, $\mathsf{causes}(q)$, and $\mathsf{confl}(\alpha)$ be as in Theorem 3. For each $\mathcal{C} \in \mathsf{causes}(q)$, consider the formulas:*

$$\varphi_{\neg \mathcal{C}} = \bigvee_{\alpha \in \mathcal{C}} \bigvee_{\substack{\beta \in \mathsf{confl}(\alpha) \\ \beta \preceq_P \alpha}} x_\beta$$

$$\varphi^{\mathcal{C}}_{\mathsf{max}} = \bigwedge_{\alpha \in R_\mathcal{C}} \Big(x_\alpha \vee \bigvee_{\substack{\beta \in \mathsf{confl}(\alpha) \\ \beta \preceq_P \alpha}} x_\beta \Big)$$

$$\varphi^{\mathcal{C}}_{\mathsf{cons}} = \bigwedge_{\substack{\alpha, \beta \in R_\mathcal{C} \\ \beta \in \mathsf{confl}(\alpha)}} \neg x_\alpha \vee \neg x_\beta$$

where $R_\mathcal{C}$ is the set of assertions reachable in $G^P_\mathcal{K}$ from some assertion β such that x_β appears in $\varphi_{\neg \mathcal{C}}$. Then $\mathcal{K} \models_{\subseteq_P\text{-IAR}} q$ iff there exists $\mathcal{C} \in \mathsf{causes}(q)$ such that the formula $\varphi_{\neg \mathcal{C}} \wedge \varphi^{\mathcal{C}}_{\mathsf{max}} \wedge \varphi^{\mathcal{C}}_{\mathsf{cons}}$ is unsatisfiable.

The above encodings can be used to answer *non-Boolean queries* using the standard reduction to the Boolean case: a tuple $\mathbf{a}$ is an answer to a non-Boolean CQ q iff the Boolean query $q[\mathbf{a}]$ is entailed under the considered semantics.

6 Experimental Evaluation

We implemented our query answering framework in Java v1.7 within our CQAPri ("Consistent Query Answering with Priorities") tool. CQAPri is built on top of the relational database server PostgreSQL v9.2.4 (`www.postgresql.org`), the Rapid v1.0 query rewriting engine for DL-Lite (Chortaras, Trivela, and Stamou 2011), and the SAT4J v2.3.4 SAT solver (Berre and Parrain 2010). All these building blocks were used with their default settings.

CQAPri stores the ABox in PostgreSQL, while it keeps the TBox in-memory together with the pre-computed set of conflicts for the KB. Conflicts are computed by evaluating

ABox id	#ABox	%conflicts	avg conflicts	o.c. graph (ms)
u1p15e-4	75708	2.05	0.04	3844
u1p5e-2	76959	30.97	1.03	4996
u1p2e-1	80454	57.99	3.96	6224
u5p15e-4	499674	1.70	0.03	19073
u5p5e-2	507713	33.12	1.21	24600
u5p2e-1	531607	58.29	4.29	32087
u10p15e-4	930729	2.37	0.05	33516
u10p5e-2	945450	33.92	1.31	43848
u10p2e-1	988882	58.89	4.86	62028
u20p15e-4	1982922	2.64	0.05	95659
u20p5e-2	2014129	33.91	1.6	122805
u20p2e-1	2103366	58.78	5.49	192450

Table 1: ABoxes in terms of size, percentage of assertions in conflicts, average number of conflicts per assertion, and time to compute the oriented conflict graph.

over the ABox the SQLized rewritings of the queries looking for counter-examples to the negative TBox inclusions. They are stored as an oriented conflict graph (Definition 7), built from a single priority level for the IAR and AR semantics, and multiple levels for the $\subseteq_P$-IAR and $\subseteq_P$-AR semantics.

When a query arrives, CQAPri evaluates it over the ABox using its SQLized rewriting, to obtain its *possible answers* and their causes. Possible answers define a superset of the answers holding under the IAR, AR, $\subseteq_P$-IAR, and $\subseteq_P$-AR semantics. Among the possible answers, CQAPri identifies the IAR ones, or an approximation of the $\subseteq_P$-IAR ones, by checking whether there is some cause whose assertions have no outgoing edges in the oriented conflict graph. For those possible answers that are not found to be IAR answers, resp. in the approximation of the $\subseteq_P$-IAR answers, deciding whether they are entailed under the AR semantics, resp. the $\subseteq_P$-IAR and $\subseteq_P$-AR semantics, is done using the SAT encodings from the preceding section.

Experimental setting

We give here a general overview of our experimental setting. More details can be found in the appendix.

TBox and Datasets We considered the modified LUBM benchmark from Lutz et al. (2013), which provides the DL-Lite$_\mathcal{R}$ version LUBM$^\exists_{20}$ of the original LUBM $\mathcal{ELI}$ TBox, and the Extended University Data Generator (EUDG) v0.1a (both available at www.informatik.uni-bremen.de/~clu/combined). We extended LUBM$^\exists_{20}$ with negative inclusions, to allow for contradictions. Inconsistencies in the ABox were introduced by contradicting the presence of an individual in a concept assertion with probability p, and the presence of each individual in a role assertion with probability $p/2$. Additionally, for every role assertion, its individuals are switched with probability $p/10$. Prioritizations of ABox were made to capture a variety of scenarios.

Table 1 describes the ABoxes we used for our experiments. Every ABox's id uXpY indicates the number X of universities generated by EUDG and the probability value Y of p for adding inconsistencies as explained above (Me-P reads $M.10^{-P}$). We chose the values for X and Y so as to have ABoxes of size varying from small to large, and a number of conflicts ranging from a value we found realistic up to values challenging our approach. We built 8 prioritizations

Query id	shape	#atoms	#vars	#rews	rew time (ms)
req2	chain	3	2	1	0
req3	dag	6	3	23	4
g2	atomic	1	1	44	0
g3	atomic	1	1	44	0
q1	chain	2	2	80	0
q2	chain	2	2	44	15
q4	dag	7	6	25	16
Lutz1	dag	8	4	3887	328
Lutz5	tree	5	3	667	16

Table 2: Queries in terms of shape, numbers of atoms variables, number of rewritings, and rewriting time (Rapid).

for each of these ABoxes further denoted by the id of the ABox it derives from, and a suffix first indicating the number of priority levels and then how these levels were chosen. lZdW indicates the number Z of priority levels: 3 and 10 in our experiments, and the distribution W: cr$^=$, a$^=$, cr$^{\neq}$, or a$^{\neq}$ indicates whether priority levels were chosen per concept/role (cr) or assertion (a), and whether choosing between these levels was equiprobable ($^=$) or not ($^{\neq}$).

Queries We used the queries described in Table 2 for our experiments. Some queries were borrowed from LUBM-based experiments found in the literature: Lutz1 and Lutz5 come from (Lutz et al. 2013), and req2 and req3 are from (Pérez-Urbina, Horrocks, and Motik 2009). The other queries we designed ourselves. They show a variety of structural aspects and rewriting sizes; they yield enough possible answers to observe the behavior of the considered semantics.

Experimental results

We report here on the general tendencies we observed. The main conclusion is that our approach scales for the considered semantics when the proportion of conflicting assertions is a few percent, as is likely to be the case in most real applications. Further details can be found in the appendix.

IAR and AR query answering We observed that the AR semantics only adds a limited number of new answers compared to the IAR semantics. For 60% of our ABox and query pairs, AR does not provide any additional answers, and it provides at most as many new answers as IAR ones.

Also, for a given number of universities (i.e., size), when the proportion of conflicting assertions increases, the number of IAR answers decreases, while the number of AR non-IAR and of possible non-AR ones increases. Such an increase significantly augments the time spent identifying AR non-IAR answers using the SAT solver, as exemplified in Figure 2 [left]. It explains that the lower the probability for generating conflicts, the more AR query answering times show a linear behavior w.r.t. ABox size (i.e., scales), up to the uXp15e-4 and uXp5e-2 ABoxes in our experiments.

$\subseteq_P$-IAR and $\subseteq_P$-AR query answering Similarly to above, the $\subseteq_P$-AR semantics adds a limited number of answers compared to the $\subseteq_P$-IAR semantics. Moreover, in most cases, the approximation of $\subseteq_P$-IAR using the ordered conflict graph identifies a large share of the $\subseteq_P$-IAR answers.

We also observed that adding prioritizations to the ABoxes complicates query answering, and using 3 priority levels typically led to harder instances than using 10 levels.

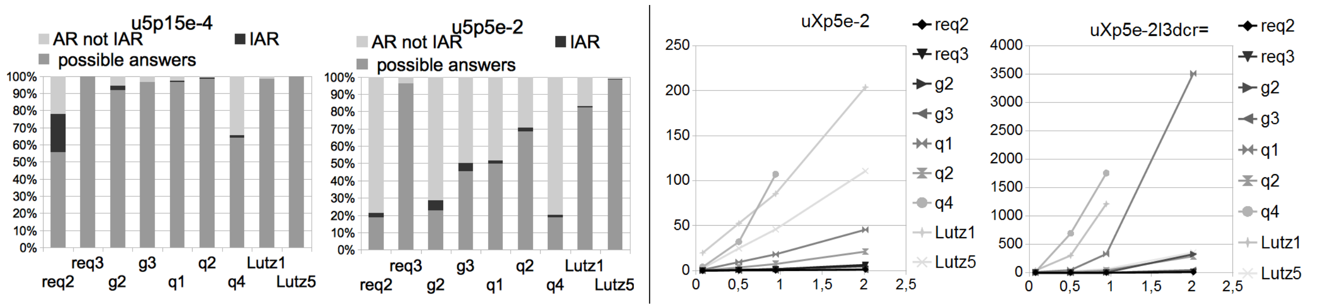

Figure 2: [left] Proportion of time spent by CQAPri to get the possible, IAR, and AR answers on two ABoxes [right] Time (in sec.) spent by CQAPri for AR and $\subseteq_P$-AR query answering on two sets of ABoxes (in millions of assertions)

Our approach scales up to the `uXp15e-4lYdZ` set of ABoxes, as answering queries against them requires in most cases less than twice the time observed for the (plain) AR semantics. Figure 2 [right] exemplifies the observed trend for the `uXp5e-2lYdZ` ABoxes, where the use of priority levels significantly increases runtimes, as well as the number of queries running out of memory or producing a time-out.

7 Related Work

The closest related work is that of (Du, Qi, and Shen 2013) who study query answering under $\leq_w$-AR semantics for the expressive DL $\mathcal{SHIQ}$. They focus on ground CQs, as such queries are better-supported by $\mathcal{SHIQ}$ reasoners. By contrast, we work with DL-Lite and can thus use query rewriting techniques to handle CQs with existential variables. We also consider IAR-based semantics and other types of preferred repairs that are not considered by Du et al.

Also related is the work on preference-based semantics for querying databases that violate integrity constraints. In (Lopatenko and Bertossi 2007), the authors study the complexity of query answering in the presence of denial constraints under the $\leq$-AR and $\leq_w$-AR semantics. Because of the difference in setting, we could not transfer their complexity results to DL-Lite. Three further preference-based semantics are proposed in (Staworko, Chomicki, and Marcinkowski 2012), based upon partially ordering the assertions that appear together in a conflict. If such an ordering is induced from an ABox prioritization, then the three semantics all coincide with our $\subseteq_P$-AR semantics.

More generally, we note that the problem of reasoning on preferred subsets has been studied in a number of other areas of AI, such as abduction, belief change, argumentation, and non-monotonic reasoning, see (Eiter and Gottlob 1995; Nebel 1998; Amgoud and Vesic 2011; Brewka, Niemelä, and Truszczynski 2008) and references therein.

Several recent works, including (Rosati 2011; Bienvenu 2012; Bienvenu and Rosati 2013), study the complexity of query answering under IAR and AR semantics. We extend this line of work by providing complexity results for variants of the IAR and AR semantics based upon preferred repairs. In some cases, we are able to adapt proof ideas from the preference-free case, but the addition of preferences also required non-trivial modifications and new ideas.

In terms of implemented tools, we are aware of two systems for inconsistency-tolerant query answering over DL KBs: the system of Du et al. (2013) for querying $\mathcal{SHIQ}$ KBs under $\leq_w$-AR semantics, and the `QuID` system (Rosati et al. 2012) that handles IAR semantics (without preferences) and DL-Lite$_\mathcal{A}$ KBs. Neither system is directly comparable to our own, since they employ different semantics. We can observe some high-level similarities with Du et al.'s system which also employs SAT solvers and uses a reachability analysis to identify a query-relevant portion of the KB.

There are also a few systems for querying inconsistent relational databases. Most relevant to our work is the `EQUIP` system (Kolaitis, Pema, and Tan 2013), which reduces AR query answering in the presence of denial constraints to binary integer programming (BIP). We considered using BIP for our own system, but our early experiments comparing the two approaches revealed better performances of the SAT-based approach on difficult problem instances.

8 Concluding Remarks

Existing inconsistency-tolerant semantics for ontology-based data access are based upon a notion of repair that makes no assumptions about the relative reliability of ABox assertions. When information on the reliability of assertions is available, it can be exploited to identify preferred repairs and improve the quality of the query results. While this idea has been explored before in various settings, there had been no systematic study of the computational properties of preferred repairs for important lightweight DLs like DL-Lite$_\mathcal{R}$. We addressed this gap in the literature by providing a thorough analysis that established the data and combined complexity of answering conjunctive and atomic queries under AR- and IAR-based semantics combined with four types of preferred repairs. Unsurprisingly, the results are mainly negative, showing that adding preferences increases complexity. However, they also demonstrate that IAR-based semantics retain some advantage over AR-based semantics and identify $\subseteq_P$-repairs as the most computationally well-behaved.

Prior work on inconsistency-tolerant querying in DL-Lite left open whether the IAR constitutes a good approximation, or whether the AR semantics can be feasibly computed in practice. Encouraged by the performance of modern-day SAT solvers and recent positive results from the database arena, we proposed a practical SAT-based approach for query answering in DL-Lite$_\mathcal{R}$ under the AR, $\subseteq_P$-IAR, and $\subseteq_P$-AR semantics, which we implemented in our CQAPri system. Our experiments show that CQAPri scales up to large ABoxes for the IAR/AR and $\subseteq_P$-IAR/$\subseteq_P$-AR semantics, when the number of conflicting assertions varies from a few percent (for all of these semantics) to a few tens of percent (only for IAR/AR). We thus show that the AR semantics *can* be computed in practice and that this is due to the fact the IAR semantics often constitutes a *very good approximation* of the AR semantics. In a similar vein, we observed that our simple approximation of the $\subseteq_P$-IAR semantics often produced a large share of the $\subseteq_P$-IAR answers, which themselves constituted a large portion of the $\subseteq_P$-AR answers.

Our long-term goal is to equip CQAPri with a portfolio of query answering techniques and an optimizer that selects the most appropriate technique for the query at hand. To this end, we have started exploring other techniques for the $\subseteq_P$-based semantics that are much more efficient on the kinds of queries easily producing timeouts or running out of memory.

Acknowledgements

The authors acknowledge Despoina Trivela for her work on an early version of the system. This work was supported by the ANR project PAGODA (ANR-12-JS02-007-01).

References

Amgoud, L., and Vesic, S. 2011. A new approach for preference-based argumentation frameworks. *Annals of Mathematics and Artificial Intelligence* 63(2):149–183.

Baader, F.; Calvanese, D.; McGuinness, D.; Nardi, D.; and Patel-Schneider, P. F., eds. 2003. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press.

Berre, D. L., and Parrain, A. 2010. The sat4j library, release 2.2. *JSAT* 7(2-3):59–64.

Bertossi, L. E. 2011. *Database Repairing and Consistent Query Answering*. Synthesis Lectures on Data Management. Morgan & Claypool Publishers.

Bienvenu, M., and Rosati, R. 2013. Tractable approximations of consistent query answering for robust ontology-based data access. In *Proc. of IJCAI*.

Bienvenu, M. 2012. On the complexity of consistent query answering in the presence of simple ontologies. In *Proc. of AAAI*.

Brewka, G.; Niemelä, I.; and Truszczynski, M. 2008. Preferences and nonmonotonic reasoning. *AI Magazine* 29(4):69–78.

Buss, S. R., and Hay, L. 1991. On truth-table reducibility to SAT. *Information and Computation* 91(1):86–102.

Calì, A.; Gottlob, G.; and Lukasiewicz, T. 2012. A general datalog-based framework for tractable query answering over ontologies. *Journal of Web Semantics* 14:57–83.

Calvanese, D.; De Giacomo, G.; Lembo, D.; Lenzerini, M.; and Rosati, R. 2007. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. *Journal of Automated Reasoning* 39(3):385–429.

Chomicki, J.; Marcinkowski, J.; and Staworko, S. 2004. Computing consistent query answers using conflict hypergraphs. In *Proc. of CIKM*, 417–426.

Chortaras, A.; Trivela, D.; and Stamou, G. 2011. Optimized query rewriting for OWL 2 QL. In *Proc. of CADE*.

Du, J.; Qi, G.; and Shen, Y.-D. 2013. Weight-based consistent query answering over inconsistent $\mathcal{SHIQ}$ knowledge bases. *Knowledge and Information Systems* 34(2):335–371.

Eiter, T., and Gottlob, G. 1995. The complexity of logic-based abduction. *Journal of the ACM* 42(1):3–42.

Eiter, T., and Gottlob, G. 1997. The complexity class Θ_2^P: Recent results and applications in AI and modal logic. In *Proc. of FCT*, 1–18.

Eiter, T.; Gottlob, G.; Ortiz, M.; and Simkus, M. 2008. Query answering in the description logic Horn-$\mathcal{SHIQ}$. In *Proc. of JELIA*, 166–179.

Hustadt, U.; Motik, B.; and Sattler, U. 2007. Reasoning in description logics by a reduction to disjunctive datalog. *Journal of Automated Reasoning* 39(3):351–384.

Kolaitis, P. G.; Pema, E.; and Tan, W.-C. 2013. Efficient querying of inconsistent databases with binary integer programming. *PVLDB* 6(6):397–408.

Krentel, M. W. 1988. The complexity of optimization problems. *Journal of Computer and System Sciences* 36(3):490–509.

Lembo, D.; Lenzerini, M.; Rosati, R.; Ruzzi, M.; and Savo, D. F. 2010. Inconsistency-tolerant semantics for description logics. In *Proc. of RR*, 103–117.

Lopatenko, A., and Bertossi, L. E. 2007. Complexity of consistent query answering in databases under cardinality-based and incremental repair semantics. In *Proc. of ICDT*, 179–193.

Lutz, C.; Seylan, I.; Toman, D.; and Wolter, F. 2013. The combined approach to OBDA: Taming role hierarchies using filters. In *Proc. of ISWC*, 314–330.

Motik, B.; Cuenca Grau, B.; Horrocks, I.; Wu, Z.; Fokoue, A.; and Lutz, C. 2012. OWL 2 Web Ontology Language profiles. W3C Recommendation. Available at `http://www.w3.org/TR/owl2-profiles/`.

Nebel, B. 1998. *Handbook of Defeasible Reasoning and Uncertainty Management Systems, Volume 3: Belief Change*. Kluwer. chapter How Hard is it to Revise a Belief Base?

Pérez-Urbina, H.; Horrocks, I.; and Motik, B. 2009. Efficient query answering for OWL 2. In *International Semantic Web Conference*, 489–504.

Rosati, R.; Ruzzi, M.; Graziosi, M.; and Masotti, G. 2012. Evaluation of techniques for inconsistency handling in OWL 2 QL ontologies. In *Proc. of ISWC*, 337–349.

Rosati, R. 2011. On the complexity of dealing with inconsistency in description logic ontologies. In *Proc. of IJCAI*, 1057–1062.

Staworko, S.; Chomicki, J.; and Marcinkowski, J. 2012. Prioritized repairing and consistent query answering in relational databases. *Annals of Mathematics and Artificial Intelligence* 64(2-3):209–246.

Wagner, K. W. 1987. More complicated questions about maxima and minima, and some closures of NP. *Theoretical Computer Science* 51:53–80.

Appendix

- Illustrative Example
- Proof of Theorem 2
- Proofs for Section 5
- Datasets
- Experimental results
- Queries and Negative Inclusions

9 Illustrative Example

We consider a simple knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ describing the university domain. The TBox

$$\mathcal{T} = \{\mathsf{Student} \sqsubseteq \mathsf{Person}, \mathsf{Professor} \sqsubseteq \mathsf{Person}, \mathsf{Student} \sqsubseteq \neg\mathsf{Professor}, \exists\mathsf{teacherOf} \sqsubseteq \mathsf{Professor}, \exists\mathsf{teacherOf}^- \sqsubseteq \mathsf{Course}, \mathsf{Course} \sqsubseteq \neg\mathsf{Person}\}$$

specifies that students and professors are persons, but students are not professors. The role teacherOf has Professor as domain and Course as range. Finally, courses are not persons. From the ABox

$$\mathcal{A} = \{\mathsf{Student}(Tom), \mathsf{Student}(Bob), \mathsf{teacherOf}(Bob, Math), \mathsf{teacherOf}(Bob, Bio), \mathsf{Student}(Joe), \mathsf{Professor}(Joe), \mathsf{teacherOf}(Bio, Joe), \mathsf{teacherOf}(Joe, Physics)\}$$

and the inclusions in $\mathcal{T}$, we can infer that Bob is both a student and professor, and Joe is a person and course.

The KB is inconsistent as both negative inclusions of $\mathcal{T}$ are violated. The repairs of $\mathcal{K}$ are:

$$\mathcal{A}'_1 = \{\mathsf{Student}(Tom), \mathsf{Student}(Bob), \mathsf{Student}(Joe)\}$$

$$\mathcal{A}'_2 = \{\mathsf{Student}(Tom), \mathsf{teacherOf}(Bob, Math), \mathsf{teacherOf}(Bob, Bio), \mathsf{Student}(Joe)\}$$

$$\mathcal{A}'_3 = \{\mathsf{Student}(Tom), \mathsf{Student}(Bob), \mathsf{Professor}(Joe), \mathsf{teacherOf}(Joe, Physics)\}$$

$$\mathcal{A}'_4 = \{\mathsf{Student}(Tom), \mathsf{teacherOf}(Bob, Math), \mathsf{teacherOf}(Bob, Bio), \mathsf{Professor}(Joe), \mathsf{teacherOf}(Joe, Physics)\}$$

$$\mathcal{A}'_5 = \{\mathsf{Student}(Tom), \mathsf{Student}(Bob), \mathsf{teacherOf}(Bio, Joe)\}$$

$$\mathcal{A}'_6 = \{\mathsf{Student}(Tom), \mathsf{teacherOf}(Bob, Math), \mathsf{teacherOf}(Bob, Bio), \mathsf{teacherOf}(Bio, Joe)\}$$

We observe that the query $\mathsf{Person}(Bob)$ is entailed under the AR semantics, since it can be inferred from every repair together with the TBox, but it is not entailed under the IAR semantics, as the interesection of the repairs does not contain any assertion concerning Bob. By contrast, the assertion $\mathsf{Student}(Tom)$ appears in every repair and hence is entailed under both the AR and IAR semantics.

By moving to preferred repair semantics, we can obtain further answers. First suppose we adopt the cardinality criterion. Then there is a single $\leq$-repair: $\mathcal{A}'_4$. Queries $\mathsf{Professor}(Bob)$ and $\mathsf{Professor}(Joe)$ are entailed under the $\leq$-IAR semantics, while they were not entailed under plain AR semantics.

Next suppose we have the following prioritization $\mathcal{P} = \langle \mathcal{P}_1, \mathcal{P}_2 \rangle$ of $\mathcal{A}$:

$$\mathcal{P}_1 = \{\mathsf{Student}(Tom), \mathsf{Student}(Bob), \mathsf{Student}(Joe), \mathsf{Professor}(Joe)\}$$

$$\mathcal{P}_2 = \{\mathsf{teacherOf}(Bob, Math), \mathsf{teacherOf}(Bob, Bio), \mathsf{teacherOf}(Joe, Physics)\}$$

The $\subseteq_P$-repairs are $\mathcal{A}'_1$ and $\mathcal{A}'_3$, and there is only one $\leq_P$-repair, namely $\mathcal{A}'_3$. Notice that $\mathsf{Student}(Bob)$ is entailed under the $\subseteq_P$-IAR and $\leq_P$-IAR semantics, whereas it was not entailed under plain AR semantics, and it conflicts with an assertion entailed under $\leq$-IAR semantics. The assertion $\mathsf{Professor}(Joe)$ is entailed under $\leq_P$-IAR semantics, but only $\mathsf{Person}(Joe)$ is entailed under $\subseteq_P$-AR semantics.

Finally, if we assign assertions in $\mathcal{P}_1$ a weight of 2, and assertions of $\mathcal{P}_2$ a weight of 1, we obtain two $\leq_w$-repairs: $\mathcal{A}'_3$ and $\mathcal{A}'_4$. Under $\leq_w$-AR semantics, neither $\mathsf{Professor}(Bob)$ nor $\mathsf{Student}(Bob)$ is entailed, but only $\mathsf{Person}(Bob)$. Under $\leq_w$-IAR semantics, $\mathsf{Professor}(Joe)$ is entailed.

10 Proof of Theorem 2

We break the proof of Theorem 2 down into several propositions. We first consider combined complexity and the AR family of semantics.

Proposition 1. *Regarding combined complexity, CQ entailment over DL-Lite KBs is Π^p_2-complete under $\leq$-AR, $\subseteq_P$-AR, $\leq_P$-AR, and $\leq_w$-AR semantics.*

Proof. First, we observe that for all four notions of preferred repair, it is possible to test in coNP whether a given set constitutes a preferred repair. Indeed, if a consistent subset of the ABox is not a repair, then this is witnessed by another consistent subset which is preferred to it. Thus, non-entailment of a CQ q from a KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ can be shown by guessing a subset $\mathcal{B} \subseteq \mathcal{A}$ and using an NP oracle to verify that (i) $\mathcal{B}$ is a preferred repair of $\mathcal{K}$, and (ii) $\langle \mathcal{T}, \mathcal{A} \rangle \not\models q$.

For the lower bounds, we note that the proof of Π^p_2-hardness of CQ entailment under plain AR semantics from (Bienvenu and Rosati 2013)can be reused for the $\leq$-AR semantics, since the $\leq$-repairs and $\subseteq$-repairs coincide for the KBs employed in that reduction. The lower bounds for the other semantics follow immediately. □

We next turn to the data complexity of query entailment under the different AR-based semantics.

Proposition 2. *Regarding data complexity, AQ and CQ entailment over DL-Lite KBs are co*NP*-complete under the $\subseteq_P$-AR semantics. For AQs, we obtain co*NP*-completeness also for combined complexity.*

Proof. We observe that it can be tested in polynomial time (w.r.t. combined complexity) whether a subset $\mathcal{B} \subseteq \mathcal{A}$ is a $\subseteq_P$-repair. This can be done by first verifying that $\mathcal{B}$ is $\mathcal{T}$-consistent and then for $1 \leq i \leq k$, checking that it is not possible to add an assertion belonging to $\mathcal{P}_i \setminus \mathcal{B}$ to $\mathcal{B} \cap (\mathcal{P}_1 \cup ... \cup \mathcal{P}_i)$ while staying $\mathcal{T}$-consistent. It follows that in the procedure from the proof of Proposition 1, properties (i) and (ii) can be verified in P w.r.t. data complexity, yielding a coNP upper bound for CQ entailment. Finally, we note that in the case of AQs, property (ii) can be checked in P w.r.t. combined complexity, and so we obtain a coNP upper bound also w.r.t. combined complexity.

The lower bound follows from the coNP-hardness of AQ entailment under the standard AR semantics (Lembo et al. 2010). □

For the $\leq_P$-AR and $\leq_w$-AR semantics, we distinguish two cases, depending on whether the maximal number of priority classes (resp. maximal weight) is considered to be fixed independently of the ABox or is counted as part of the input. The former assumption is made in (Du, Qi, and Shen 2013), where a $\Delta_2^p[O(\log n)]$ upper bound is given for the expressive DL $\mathcal{SHIQ}$ that contains DL-Lite$_\mathcal{R}$ as a fragment. Du et al. 2013 also give a $\Delta_2^p[O(\log n)]$ lower bound for atomic queries, but the proof uses constructs that are unavailable in DL-Lite. Likewise, the proof of $\Delta_2^p[O(\log n)]$-hardness for atomic queries under the $\leq$-AR semantics from (Lopatenko and Bertossi 2007) utilizes denial constraints that cannot be expressed in DL-Lite.

Proposition 3. *Regarding data complexity, AQ and CQ entailment over DL-Lite KBs are:*

- Δ_2^p*-complete for the* $\leq_P$*-AR and* $\leq_w$*-AR semantics,*
- $\Delta_2^p[O(\log n)]$*-complete for the* $\leq$*-AR semantics, and for the* $\leq_P$*-AR and* $\leq_w$*-AR semantics, if there is an ABox-independent bound on the number of priority classes (resp. maximal weight).*

For AQs, these results also hold w.r.t. combined complexity.

For the proof of Proposition 3 and later propositions, we will leverage the following result, which demonstrates how the semantics based upon the prioritized cardinality preference relation can be recast in terms of weight functions.

Lemma 1. *(Adapted from (Eiter and Gottlob 1995)) Let* $P = \langle \mathcal{P}_1, \dots, \mathcal{P}_k \rangle$ *be a prioritization of* $\mathcal{A}$*, let* $u = (\mathsf{max}_{i=1}^k |\mathcal{P}_i|) + 1$*, and let* w *be defined by:* $w(\alpha) = u^{k-i}$ *for* $\alpha \in \mathcal{P}_i$*. Then the set of* $\leq_P$*-repairs of* $\langle \mathcal{T}, \mathcal{A} \rangle$ *coincides with the set of* $\leq_w$*-repairs of* $\langle \mathcal{T}, \mathcal{A} \rangle$*, for every TBox* $\mathcal{T}$*.*

The proof of Proposition 3 will also make frequent use of the following characterization of $\leq$-repairs.

Lemma 2. *A subset* $\mathcal{A}'$ *of* $\mathcal{A}$ *is a* $\leq$*-repair of* $\mathcal{A}$ *w.r.t.* $\mathcal{T}$ *if and only if* $\mathcal{A}'$ *is* $\mathcal{T}$*-consistent and there do not exist subsets* X *of* $\mathcal{A}'$ *and* Y *of* $\mathcal{A} \backslash \mathcal{A}'$ *such that* $|Y| > |X|$ *and* $(\mathcal{A}' \setminus X) \cup Y$ *is* $\mathcal{T}$*-consistent.*

Proof. For the first direction, let $\mathcal{A}'$ be in $Rep_{\leq}(\mathcal{K})$. Suppose for a contradiction that there exist a subset X of $\mathcal{A}'$ and a subset Y of $\mathcal{A} \backslash \mathcal{A}'$ such that $|Y| > |X|$ and $(\mathcal{A}' \setminus X) \cup Y$ is $\mathcal{T}$-consistent. Let $\mathcal{A}'' = (\mathcal{A}' \setminus X) \cup Y$. Then $\mathcal{A}''$ is a $\mathcal{T}$-consistent subset of $\mathcal{A}$ and $|\mathcal{A}''| = |\mathcal{A}'| - |X| + |Y|$, so $|\mathcal{A}''| > |\mathcal{A}'|$. Hence, $\mathcal{A}'$ is not a $\leq$-repair.

For the other direction, let $\mathcal{A}'$ be a $\mathcal{T}$-consistent subset of $\mathcal{A}$ such that there does not exist any subset X of $\mathcal{A}'$ such that there exists a subset Y of $\mathcal{A} \backslash \mathcal{A}'$ such that $|Y| > |X|$ and $(\mathcal{A}' \setminus X) \cup Y$ is $\mathcal{T}$-consistent. Suppose for a contradiction that $\mathcal{A}' \notin Rep_{\leq}(\mathcal{K})$. Let $\mathcal{A}'' \in Rep_{\leq}(\mathcal{K})$. Since $\mathcal{A}'$ is not a $\leq$-repair, $|\mathcal{A}''| > |\mathcal{A}'|$. Let $X = \mathcal{A}' \backslash \mathcal{A}''$ and $Y = \mathcal{A}'' \backslash \mathcal{A}'$. Then $(\mathcal{A}' \setminus X) \cup Y = \mathcal{A}''$ is $\mathcal{T}$-consistent and $|Y| = |\mathcal{A}''| - |\mathcal{A}'' \cap \mathcal{A}'|$ and $|X| = |\mathcal{A}'| - |\mathcal{A}' \cap \mathcal{A}''|$, so $|Y| > |X|$. Hence X, Y contradict the assumption. It follows that $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$. □

We are now ready to give the proof of Proposition 3.

Proof of Proposition 3.

Upper bounds. For the $\leq_w$-AR semantics, we use the following procedure to decide whether $\mathcal{K} \not\models_{\leq_w\text{-AR}} q$:

1. Compute the weight u_{rep} of $\leq_w$-repairs by binary search, calling the NP oracle to determine whether there exists a $\mathcal{T}$-consistent subset $\mathcal{A}' \subseteq \mathcal{A}$ such that $\sum_{\alpha \in \mathcal{A}'} w(\alpha) \geq u$ where u is the input.
2. Call the NP oracle to determine whether there exists $\mathcal{B} \subseteq \mathcal{A}$, $\mathcal{T}$-consistent and such that $\sum_{\alpha \in \mathcal{B}} w(\alpha) = u_{\text{rep}}$ and $\langle \mathcal{B}, \mathcal{T} \rangle \not\models q$. Return "not entailed" if the call succeeds.

This procedure yields membership in Δ_2^p for the general case. If there is an ABox-independent bound b on the maximal value of w, then $u_{\text{rep}} \leq \sum_{\alpha \in \mathcal{A}} w(\alpha) \leq b|\mathcal{A}|$, where b is treated as a constant. It follows that the procedure requires only logarithmically many oracle calls, yielding an improved upper bound of $\Delta_2^p[O(\log n)]$. Note that in the case of AQs, we can test $\langle \mathcal{B}, \mathcal{T} \rangle \not\models q$ in polynomial time w.r.t. combined complexity, so these upper bounds apply also w.r.t. combined complexity.

We can derive the upper bounds for the $\leq_P$-AR semantics by applying Lemma 1. Note that when there is a bound k on the number of priority levels, then this implies a polynomial bound of $(\mathsf{max}_{i=1}^k |\mathcal{P}_i| + 1)^{k-1} \leq (|\mathcal{A}| + 1)^{k-1}$ on the maximal weight of the corresponding weight function, and so the $\Delta_2^p[O(\log n)]$ upper bound applies. This holds in particular when $k = 1$, i.e., for the $\leq$-AR semantics.

Δ_2^p lower bound for AQs under $\leq_P$-AR semantics. The proof is by reduction from the following Δ_2^P-complete problem, cf. (Krentel 1988): given a satisfiable 3CNF formula $\varphi = c_1 \wedge \ldots \wedge c_m$ over variables $x_1, ..., x_n$, decide whether the lexicographically maximum truth assignment satisfying φ with respect to $(x_1, ..., x_n)$, denoted by $\nu_{\max}$, fulfills $\nu_{\max}(x_n) = \mathtt{true}$. We encode this problem as a $\leq_P$-AR query entailment problem as follows:

$$\begin{aligned}
\mathcal{T} = &\{T \sqsubseteq \neg\exists N_\ell^- \mid 1 \leq \ell \leq 3\} \cup \\
&\{\exists P_\ell \sqsubseteq \neg\exists N_{\ell'}, \exists P_\ell^- \sqsubseteq \neg\exists N_{\ell'}^- \mid 1 \leq \ell, \ell' \leq 3\} \cup \\
&\{\exists P_\ell \sqsubseteq \neg\exists P_{\ell'}, \exists N_\ell \sqsubseteq \neg\exists N_{\ell'} \mid 1 \leq \ell \neq \ell' \leq 3\} \\
\mathcal{A} = &\{T(x_i) \mid 1 \leq i \leq n\} \cup \\
&\{P_\ell(c_j, x_i) \mid x_i \text{ is the } \ell^{th} \text{ literal of } c_j\} \cup \\
&\{N_\ell(c_j, x_i) \mid \neg x_i \text{ is the } \ell^{th} \text{ literal of } c_j\} \\
q = &\, T(x_n)
\end{aligned}$$

with the prioritization $P = \langle \mathcal{P}_1, ..., \mathcal{P}_{n+1} \rangle$ of $\mathcal{A}$ as follows: $\mathcal{P}_1 = \mathcal{A} \setminus \{T(x_i) \mid 1 \leq i \leq n\}$, and for $1 < i \leq n+1$, $\mathcal{P}_i = \{T(x_{i-1})\}$. One can show that $\langle \mathcal{T}, \mathcal{A} \rangle \models_{\leq_P\text{-}AR} q$ iff $\nu_{\max}(x_n) = \mathtt{true}$.

$\Delta_2^p[O(\log n)]$-hardness for AQs under $\leq$-AR semantics. The proof is by reduction from the Parity(3SAT) problem, cf. (Wagner 1987; Eiter and Gottlob 1997). We recall that a Parity(3SAT) instance is given by a sequence $\varphi_1, \dots, \varphi_n$ of propositional formulas in 3CNF, and the problem is to decide whether the number of satisfiable formulas is odd. It is known that it can be assumed w.l.o.g. that the formulas are such that φ_{i+1} is unsatisfiable whenever φ_i is unsatisfiable. Consequently, the problem reduces to deciding existence of an odd integer k such that φ_k is satisfiable and φ_{k+1} is unsatisfiable.

Consider a Parity(3SAT) instance given by $\varphi_1, \dots, \varphi_n$. For each i, $1 \leq i \leq n$, let $\{c_{i,1}, ..., c_{i,m(i)}\}$ be the clauses of φ_i over variables $X_i = \{x_{i,1}, ..., x_{i,l(i)}\}$. We define an $\leq$-AR query entailment problem as follows:

$$\begin{aligned}
\mathcal{T} =& \{\exists A \sqsubseteq Y\} \cup \\
& \{\exists A^- \sqsubseteq \neg V^d, \exists A^- \sqsubseteq \neg W^d \mid 1 \leq d \leq 3\} \cup \\
& \{W^d \sqsubseteq \neg \exists K^m \mid 1 \leq d \leq 3, 1 \leq m \leq 4\} \cup \\
& \{V^d \sqsubseteq \neg \exists E^m \mid 1 \leq d \leq 3, 1 \leq m \leq 4\} \cup \\
& \{\exists K^{m-} \sqsubseteq \neg \exists F^r \mid 1 \leq m \leq 4, 1 \leq r \leq 7\} \cup \\
& \{\exists E^{m-} \sqsubseteq \neg S^g \mid 1 \leq m \leq 4, 1 \leq g \leq 5\} \cup \\
& \{S^g \sqsubseteq \neg \exists F^r \mid 1 \leq g \leq 5, 1 \leq r \leq 7\} \cup \\
& \{\exists F^{r-} \sqsubseteq \neg \exists P^k_l \mid 1 \leq r \leq 7, 1 \leq k \leq 8, 1 \leq l \leq 3\} \cup \\
& \{\exists F^{r-} \sqsubseteq \neg \exists N^k_l \mid 1 \leq r \leq 7, 1 \leq k \leq 8, 1 \leq l \leq 3\} \cup \\
& \{\exists N^{k-}_l \sqsubseteq \neg \exists P^{k'-}_{l'} \mid 1 \leq k, k' \leq 8, 1 \leq l, l' \leq 3\} \cup \\
& \{\exists N^k_l \sqsubseteq \neg \exists P^{k'}_{l'} \mid 1 \leq k, k' \leq 8, 1 \leq l, l' \leq 3\} \cup \\
& \{\exists P^k_l \sqsubseteq \neg \exists P^{k'}_{l'} \mid 1 \leq k, k' \leq 8, 1 \leq l \neq l' \leq 3\} \cup \\
& \{\exists N^k_l \sqsubseteq \neg \exists N^{k'}_{l'} \mid 1 \leq k, k' \leq 8, 1 \leq l \neq l' \leq 3\} \\
\mathcal{A} =& \{A(y, a_i) \mid i \equiv 1 \bmod 2, 1 \leq i \leq n\} \\
& \cup \{K^m(a_i, \varphi_i), W^d(a_i) \mid i \equiv 1 \bmod 2, \\
& \quad 1 \leq i \leq n, 1 \leq d \leq 3, 1 \leq m \leq 4\} \\
& \cup \{E^m(a_{i-1}, \varphi_i), V^d(a_{i-1}) \mid i \equiv 0 \bmod 2, \\
& \quad 1 \leq i \leq n, 1 \leq d \leq 3, 1 \leq m \leq 4\} \\
& \cup \{S^g(\varphi_i) \mid 1 \leq i \leq n, 1 \leq g \leq 5\} \\
& \cup \{F^r(\varphi_i, c_{i,j}) \mid 1 \leq i \leq n, 1 \leq j \leq m(i), 1 \leq r \leq 7\} \\
& \cup \{P^k_l(c_{i,j}, x_{i,h}) \mid x_{i,h} \text{ is the } l^{th} \text{ literal of } c_{i,j}, \\
& \quad 1 \leq i \leq n, 1 \leq l \leq 3, 1 \leq k \leq 8\} \\
& \cup \{N^k_l(c_{i,j}, x_{i,h}) \mid \neg x_{i,h} \text{ is the } l^{th} \text{ literal of } c_{i,j}, \\
& \quad 1 \leq i \leq n, 1 \leq l \leq 3, 1 \leq k \leq 8\} \\
q =& Y(y)
\end{aligned}$$

Note that $\mathcal{T}$, $\mathcal{A}$ and q can be constructed in time polynomial in $\varphi_1, ..., \varphi_n$.

First, notice that if $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$ and $C^p(x) \in \mathcal{A}'$ with $C \in \{V, W, S\}$, then $C^{p'}(x) \in \mathcal{A}'$ for all $C^{p'}(x) \in \mathcal{A}$. Indeed, since $C^p(x)$ and $C^{p'}(x)$ do not conflict with each other and they conflict with the same assertions of $\mathcal{A}$, both or neither will appear in $\mathcal{A}'$. For the same reasons, if $R^p(x, z) \in \mathcal{A}'$ with $R \in \{K, E, F, P, N\}$, then $R^{p'}(x, z) \in \mathcal{A}'$ for every assertion $R^{p'}(x, z) \in \mathcal{A}$. Hence, if $F^1(\varphi_i, c_{i,j}) \in \mathcal{A}'$ for instance, then the seven assertions $F^r(\varphi_i, c_{i,j})$ $(1 \leq r \leq 7)$ are in $\mathcal{A}'$, and if $F^1(\varphi_i, c_{i,j}) \notin \mathcal{A}'$, then no assertion of the form $F^r(\varphi_i, c_{i,j})$ belongs to $\mathcal{A}'$.

Next we establish a series of claims that further characterize the sets in $Rep_{\leq}(\mathcal{K})$.

Claim 1 If φ_i is satisfiable and $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$, then $S^1(\varphi_i) \in \mathcal{A}'$.

Proof of claim. Suppose that φ_i is satisfiable and let $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$. Since φ_i is satisfiable, there exists a truth assignment $\Phi_i(X_i)$ such that $\nu_{\Phi_i}(\varphi_i) = \mathsf{true}$. It follows that for every clause $c_{i,j}$ of φ_i, $\nu_{\Phi_i}(c_{i,j}) = \mathsf{true}$ so there exists k such that $x_{i,k} \in c_{i,j}$ and $\Phi_i(x_{i,k}) = \mathsf{true}$ or $\neg x_{i,k} \in c_{i,j}$ and $\Phi_i(x_{i,k}) = \mathsf{false}$. Let

$$\begin{aligned}
\mathcal{A}'_{\Phi_i} =& \{S^g(\varphi_i) \mid 1 \leq g \leq 5\} \\
& \cup \{P^k_l(c_{i,j}, x_{i,h}) \mid x_{i,h}\ l^{th} \text{ literal of } c_{i,j}, \\
& \qquad \Phi_i(x_{i,h}) = \mathsf{true}, 1 \leq k \leq 8\} \\
& \cup \{N^k_l(c_{i,j}, x_{i,h}) \mid \neg x_{i,h}\ l^{th} \text{ literal of } c_{i,j}, \\
& \qquad \Phi_i(x_{i,h}) = \mathsf{false}, 1 \leq k \leq 8\}
\end{aligned}$$

and $\mathcal{A}_{\Phi_i}$ be a $\subseteq$-repair of $\mathcal{A}'_{\Phi_i}$ w.r.t. $\mathcal{T}$. By construction, the conflicts of $\mathcal{A}'_{\Phi_i}$ are of the form $\{P^k_l(c_{i,j}, x_{i,h}), P^{k'}_{l'}(c_{i,j}, x_{i,h})\}$,$\{N^k_l(c_{i,j}, x_{i,h}), N^{k'}_{l'}(c_{i,j}, x_{i,h})\}$ or $\{P^k_l(c_{i,j}, x_{i,h}), N^{k'}_{l'}(c_{i,j}, x_{i,h})\}$. Hence $S^g(\varphi_i) \in \mathcal{A}_{\Phi_i}$ $(1 \leq g \leq 5)$ and for each clause $c_{i,j}$, there exists exactly one h such that $P^k_l(c_{i,j}, x_{i,h}) \in \mathcal{A}_{\Phi_i}$ or $N^k_l(c_{i,j}, x_{i,h}) \in \mathcal{A}_{\Phi_i} (1 \leq k \leq 8)$ (otherwise $\mathcal{A}_{\Phi_i}$ would not be maximal for set inclusion).

Suppose for a contradiction that $S^1(\varphi_i) \notin \mathcal{A}'$ (thus $S^g(\varphi_i) \notin \mathcal{A}'$, $1 \leq g \leq 5$). Let $Y = \mathcal{A}_{\Phi_i} \backslash \mathcal{A}'$ and let X be the set of the assertions of $\mathcal{A}'$ which conflict with some assertion of Y w.r.t. $\mathcal{T}$. By construction of X, $(\mathcal{A}' \backslash X) \cup Y$ is $\mathcal{T}$-consistent. Since $S^g(\varphi_i) \in Y, 1 \leq g \leq 5$, and letting n_c be the number of clauses $c_{i,j}$ such that $P^k_l(c_{i,j}, x_{i,h}) \in Y$ or $N^k_l(c_{i,j}, x_{i,h}) \in Y (1 \leq k \leq 8)$, $|Y| = 5 + n_c * 8$. X can contain at most 4 assertions of the form $E^m(a_{i-1}, \varphi_i)$ (if i is even), which conflict with the $S^g(\varphi_i)$, and n_c sets of 7 $F^r(\varphi_i, c_{i,j})$ or of 8 $N^k_l(c_{i,j}, x_{i,h}) \notin Y$ or 8 $P^k_l(c_{i,j}, x_{i,h}) \notin Y$, which conflict with the $P^{k'}_{l'}(c_{i,j}, x_{i,h}) \in Y$ or $N^{k'}_{l'}(c_{i,j}, x_{i,h}) \in Y (1 \leq k' \leq 8)$. Hence $|X| \leq 4 + n_c * 8$. It follows that $|X| < |Y|$, so applying Lemma 2, we get $\mathcal{A}' \notin Rep_{\leq}(\mathcal{K})$. *(End proof of Claim 1)*

Claim 2 If there exists $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$ such that $S^1(\varphi_i) \in \mathcal{A}'$, then φ_i is satisfiable.

Proof of claim. Let $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$ be such that $S^1(\varphi_i) \in \mathcal{A}'$. For all $c_{i,j}$ and $1 \leq r \leq 7$, $F^r(\varphi_i, c_{i,j}) \notin \mathcal{A}'$.

Suppose for a contradiction that there exists $c_{i,j}$ such that for all $x_{i,h}$ and all l, $P^1_l(c_{i,j}, x_{i,h}) \notin \mathcal{A}'$, $N^1_l(c_{i,j}, x_{i,h}) \notin \mathcal{A}'$. If i is even, it is possible to add 7 assertions of the form $F^r(\varphi_i, c_{i,j})$ and remove the 5 assertions $S^g(\varphi_i) \in \mathcal{A}'$ to obtain a $\mathcal{T}$-consistent subset of $\mathcal{A}$, contradicting the fact that $\mathcal{A}'$ is a $\leq$-repair. If i is odd, it is possible to add 7 assertions of the form $F^r(\varphi_i, c_{i,j})$ and 3 assertions of the form $W^d(a_i)$ and remove the 5 assertions $S^g(\varphi_i) \in \mathcal{A}'$ and 4 assertions $K^m(a_i, \varphi_i) \in \mathcal{A}'$ (or add 7 assertions of the form $F^r(\varphi_i, c_{i,j})$ and remove the 5 assertions $S^g(\varphi_i) \in \mathcal{A}'$ if there is no assertions of the form $K^m(a_i, \varphi_i)$ in $\mathcal{A}'$) to obtain a $\mathcal{T}$-consistent subset of $\mathcal{A}$, contradicting the fact that $\mathcal{A}'$ is a $\leq$-repair. Thus for all $c_{i,j}$, there exists $x_{i,h}$ such that $P^1_l(c_{i,j}, x_{i,h}) \in \mathcal{A}'$ or $N^1_l(c_{i,j}, x_{i,h}) \in \mathcal{A}'$.

Let $\Phi_i(X_i)$ be the truth assignment defined as follows:

- $\Phi_i(x_{i,h}) = \mathsf{true}$ if there exists some assertion $P^1_l(c_{i,j}, x_{i,h}) \in \mathcal{A}'$
- $\Phi_i(x_{i,h}) = \mathsf{false}$ if there exists some assertion $N^1_l(c_{i,j}, x_{i,h}) \in \mathcal{A}'$

- $\Phi_i(x_{i,h}) = \mathsf{true}$ otherwise

By construction of Φ_i, $\nu_{\Phi_i}(c_{i,j}) = \mathsf{true}$ for every clause $c_{i,j}$ of φ_i. It follows that φ_i is satisfiable. *(End proof of Claim 2)*

Claim 3 If φ_i is unsatisfiable and $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$, then there exists j such that $F^1(\varphi_i, c_{i,j}) \in \mathcal{A}'$.

Proof of claim. Suppose that φ_i is unsatisfiable and let $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$. Since φ_i is unsatisfiable, $S^g(\varphi_i) \notin \mathcal{A}'$, $1 \leq g \leq 5$ (by 2). Hence there exists j such that $F^r(\varphi_i, c_{i,j}) \in \mathcal{A}'$ $(1 \leq r \leq 7)$ or $E^m(a_{i-1}, \varphi_i) \in \mathcal{A}'$ $(1 \leq m \leq 4)$ (otherwise $\mathcal{A}'$ would not be maximal).

Thus, if i is odd, then there exists j such that $F^r(\varphi_i, c_{i,j}) \in \mathcal{A}'$ $(1 \leq r \leq 7)$.

If i is even, suppose for a contradiction that $F^r(\varphi_i, c_{i,j}) \notin \mathcal{A}'$ $(1 \leq r \leq 7)$ for every j. Thus $E^m(a_{i-1}, \varphi_i) \in \mathcal{A}'$ $(1 \leq m \leq 4)$ so $V^d(a_{i-1}) \notin \mathcal{A}'$ $(1 \leq d \leq 3)$. Let $X = (\{A(y, a_i)\} \cup \{E^m(a_{i-1}, \varphi_i) \mid 1 \leq m \leq 4\}) \cap \mathcal{A}'$ and $Y = \{S^g(\varphi_i) | 1 \leq g \leq 5\} \cup \{V^d(a_{i-1}) \mid 1 \leq d \leq 3\}$. $|X| \leq 5$ and $|Y| = 8$. $X \subseteq \mathcal{A}'$, $Y \subseteq \mathcal{A} \backslash \mathcal{A}'$, $|Y| > |X|$ and $(\mathcal{A}' \backslash X) \cup Y$ is $\mathcal{T}$-consistent so $\mathcal{A}'$ is not a $\leq$-repair.

In both cases, there exists j such that $F^r(\varphi_i, c_{i,j}) \in \mathcal{A}'$ $(1 \leq r \leq 7)$. *(End proof of Claim 3)*

Claim 4 If there exists $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$ such that there exists j such that $F^1(\varphi_i, c_{i,j}) \in \mathcal{A}'$, then φ_i is unsatisfiable.

Proof of claim. Let $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$ be such that there exists j such that $F^1(\varphi_i, c_{i,j}) \in \mathcal{A}'$. Suppose for a contradiction that φ_i is satisfiable. Then $S^1(\varphi_i) \in \mathcal{A}'$ (by 1) which is not possible since $F^1(\varphi_i, c_{i,j})$ and $S^1(\varphi_i)$ are in a conflict. (*End proof of Claim 4*)

Claim 5 Let $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$. If there exists an odd integer k such that $S^1(\varphi_k) \in \mathcal{A}'$ and there exists j such that $F^1(\varphi_{k+1}, c_{k+1,j}) \in \mathcal{A}'$, or such that $S^1(\varphi_k) \in \mathcal{A}'$ and $k = n$, then $A(y, a_k) \in \mathcal{A}'$.

Proof of claim. Let $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$ satisfy the above conditions, and suppose for a contradiction that $A(y, a_k) \notin \mathcal{A}'$. Since $\mathcal{A}'$ is maximal, $W^d(a_k) \in \mathcal{A}'$ $(1 \leq d \leq 3)$ or $V^d(a_k) \in \mathcal{A}'$ $(1 \leq d \leq 3)$. Hence, if $k = n$, then $K^m(a_k, \varphi_k) \notin \mathcal{A}'$ $(1 \leq m \leq 4)$, and if $k \neq n$, then $K^m(a_k, \varphi_k) \notin \mathcal{A}'$ $(1 \leq m \leq 4)$ or $E^m(a_k, \varphi_{k+1}) \notin \mathcal{A}'$ $(1 \leq m \leq 4)$.

Let $X = (\{W^d(a_k) \mid 1 \leq d \leq 3\} \cup \{V^d(a_k) | 1 \leq d \leq 3\}) \cap \mathcal{A}'$ and $Y = (\{K^m(a_k, \varphi_k) \mid 1 \leq m \leq 4\} \cup \{E^m(a_k) \mid 1 \leq m \leq 4\}) \cap (\mathcal{A} \backslash \mathcal{A}')$. If $k = n$ then $|X| \leq 3$ and $|Y| = 4$. If $k \neq n$ then $|Y| \geq 4$ and if $4 \leq |X| \leq 6$ then $|Y| = 8$ (since in this case X contains assertions of the form $W^d(a_k)$ and $V^d(a_k)$). In both cases, we have $X \subseteq \mathcal{A}'$, $Y \subseteq \mathcal{A} \backslash \mathcal{A}'$, $|Y| > |X|$ and $(\mathcal{A}' \backslash X) \cup Y$ is $\mathcal{T}$-consistent, so $\mathcal{A}'$ is not a $\leq$-repair.

We have obtained the desired contradiction, so we can conclude that $A(y, a_k) \in \mathcal{A}'$. (*End proof of Claim 5*)

Claim 6 Let $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$. If there exists an odd integer k such that $A(y, a_k) \in \mathcal{A}'$, then $S^1(\varphi_k) \in \mathcal{A}'$ and there exists j such that $F^1(\varphi_{k+1}, c_{k+1,j}) \in \mathcal{A}'$, or $S^1(\varphi_k) \in \mathcal{A}'$ and $k = n$.

Proof of Claim 6. Let $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$ with $A(y, a_k) \in \mathcal{A}'$, for k an odd integer.

First suppose for a contradiction that $S^1(\varphi_k) \notin \mathcal{A}'$. Since k is odd, $S^1(\varphi_k)$ cannot be in a conflict with some $E^m(a_k)$, so there exists l such that $F^r(\varphi_k, c_{k,l}) \in \mathcal{A}'$ $(1 \leq r \leq 7)$. Hence $K^m(a_k, \varphi_k) \notin \mathcal{A}'$ $(1 \leq m \leq 4)$. Let $X = \{A(y, a_k)\}$ and $Y = \{W^d(a_k) \mid 1 \leq d \leq 3\}$. As $X \subseteq \mathcal{A}'$, $Y \subseteq \mathcal{A} \backslash \mathcal{A}'$, $|Y| > |X|$ and $(\mathcal{A}' \backslash X) \cup Y$ is $\mathcal{T}$-consistent, it follows that $\mathcal{A}'$ is not a $\leq$-repair. This is a contradiction, so we may conclude that $S^1(\varphi_k) \in \mathcal{A}'$.

If $k \neq n$, suppose for a contradiction that for every j, $F^r(\varphi_{k+1}, c_{k+1,j}) \notin \mathcal{A}'$ $(1 \leq r \leq 7)$. Since $A(y, a_k) \in \mathcal{A}'$, $V^d(a_k) \notin \mathcal{A}'$ $(1 \leq d \leq 3)$ so $S^g(\varphi_{k+1}) \in \mathcal{A}'$ $(1 \leq g \leq 5)$ and $E^m(a_k, \varphi_{k+1}) \notin \mathcal{A}'$ $(1 \leq m \leq 4)$ (otherwise removing the 4 $E^m(a_k, \varphi_{k+1})$ and adding the 5 $S^g(\varphi_{k+1})$ will provide a $\mathcal{T}$-consistent subset of $\mathcal{A}$ with a greater cardinality than $\mathcal{A}'$). Let $X = \{A(y, a_k)\}$ and $Y = \{V^d(a_k) \mid 1 \leq d \leq 3\}$. $X \subseteq \mathcal{A}'$, $Y \subseteq \mathcal{A} \backslash \mathcal{A}'$, $|Y| > |X|$ and $(\mathcal{A}' \backslash X) \cup Y$ is $\mathcal{T}$-consistent, so $\mathcal{A}'$ is not a $\leq$-repair. Again we have a reached a contradiction, and so may infer that there is some j such that $F^r(\varphi_{k+1}, c_{k+1,j}) \in \mathcal{A}'$ $(1 \leq r \leq 7)$. (*End proof of Claim 6*)

We are now ready to show that $\mathcal{K} \models_{\leq\text{-AR}} q$ if and only if the answer of the initial parity SAT problem is "yes".

- First suppose that there exists an odd integer k such that φ_k is satisfiable and φ_{k+1} is unsatisfiable (or $k = n$). Let $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$. Since φ_k is satisfiable, $S^1(\varphi_k) \in \mathcal{A}'$ (by Claim 1), and since φ_{k+1} is unsatisfiable (or $k = n$) there exists j such that $F^1(\varphi_{k+1}, c_{k+1,j}) \in \mathcal{A}'$ (by Claim 3) (or $k = n$). Hence $A(y, a_k) \in \mathcal{A}'$ (by Claim 5). Thus for every $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$, $A(y, a_k) \in \mathcal{A}'$ so $\langle \mathcal{A}', \mathcal{T} \rangle \models Y(y)$. Hence $\mathcal{K} \models_{\leq\text{-AR}} q$.
- To show the other direction, suppose that $\mathcal{K} \models_{\leq\text{-AR}} Y(y)$, and let $\mathcal{A}' \in Rep_{\leq}(\mathcal{K})$. Then $\langle \mathcal{A}', \mathcal{T} \rangle \models Y(y)$, so there exists an (necessarily odd) integer k such that $A(y, a_k) \in \mathcal{A}'$ Thus by Claim 6, $S^1(\varphi_k) \in \mathcal{A}'$ and there exists j such that $F^1(\varphi_{k+1}, c_{k+1,j}) \in \mathcal{A}'$, or $S^1(\varphi_k) \in \mathcal{A}'$ and $k = n$. It follows that φ_k is satisfiable (by Claim 2) and φ_{k+1} is unsatisfiable (by Claim 4) or φ_k is satisfiable and $k = n$.

Remaining lower bounds. Applying Lemma 1, we can transfer the Δ_2^p lower bound for AQs under $\leq_P$-AR semantics to the $\leq_w$-AR semantics. The preceding $\Delta_2^p[O(\log \mathrm{n})]$ lower bound for AQs under the $\leq$-AR semantics transfers to the $\leq_P$-AR semantics (under the bounded priority level assumption). The latter result can then be transferred using Lemma 1 to the $\leq_w$-AR semantics (under the bounded weight assumption). ☐

We next establish the complexity of query entailment under the different IAR semantics.

Proposition 4. *Regarding data complexity, AQ and CQ entailment over DL-Lite KBs is* coNP*-complete for the* $\subseteq_P$*-IAR semantics. For AQs, we also have* coNP*-complete regarding combined complexity.*

Proof. We can show that $\langle \mathcal{T}, \mathcal{A} \rangle \not\models_{\subseteq_P\text{-IAR}} q$ as follows:

1. Guess a subset $\mathcal{A}' = \{\alpha_1, \ldots, \alpha_m\} \subseteq \mathcal{A}$ together with a subset $\mathcal{B}_i \subseteq \mathcal{A}$ with $\alpha_i \notin \mathcal{B}_i$ for each $1 \leq i \leq m$.
2. Verify that (i) each $\mathcal{B}_i$ is a $\subseteq_P$-repair and (ii) $\langle \mathcal{T}, \mathcal{A} \setminus \mathcal{A}' \rangle \not\models q$.

Since $\subseteq_P$-repairs can be identified in polynomial time (cf. proof of Proposition 2) and query entailment is in P for data complexity (Theorem 1), the above procedure runs in non-deterministic polynomial time in the size of $\mathcal{A}$. We thus obtain a coNP upper bound for data complexity. For AQs, query entailment is in P for combined complexity, so we obtain an coNP upper bound also for combined complexity.

We show coNP-hardness using a variant of a reduction from UNSAT from (Bienvenu 2012). Let $\varphi = c_1 \wedge \ldots \wedge c_m$ be a propositional CNF with variables $x_1, ..., x_n$. Consider the TBox and prioritized ABox defined as follows:

$$\begin{aligned}\mathcal{T} = &\{\exists P^- \sqsubseteq \neg\exists N^-, \exists P \sqsubseteq \neg\exists U^-, \exists N \sqsubseteq \neg\exists U^-,\\ &\exists U \sqsubseteq \neg B, B \sqsubseteq \neg A\}\\ \mathcal{P}_1 = &\{P(c_j, x_i) \mid 1 \leq j \leq m, x_i \in c_j\} \cup\\ &\{N(c_j, x_i) \mid 1 \leq j \leq m, \neg x_i \in c_j\}\\ \mathcal{P}_2 = &\{U(a, c_j) \mid 1 \leq j \leq m\}\\ \mathcal{P}_3 = &\{A(a), B(a)\}\end{aligned}$$

with $\mathcal{A} = \mathcal{P}_1 \cup \mathcal{P}_2 \cup \mathcal{P}_3$ and $P = \langle \mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3 \rangle$. It can be verified that $\langle \mathcal{T}, \mathcal{A} \rangle \models_{\subseteq_P\text{-IAR}} A(a)$ iff φ is unsatisfiable. □

Proposition 5. *For data and combined complexity, AQ and CQ entailment over DL-Lite KBs is:*

- Δ_2^p*-complete for the* $\leq_P$*-IAR and* $\leq_w$*-IAR semantics,*
- $\Delta_2^p[O(\log n)]$*-complete for the* $\leq$*-IAR semantics, and for the* $\leq_P$*-IAR and* $\leq_w$*-IAR semantics, if there is an ABox-independent bound on the number of priority classes (resp. maximal weight).*

We also have $\Delta_2^p[O(\log n)]$*-completeness for CQ entailment under* $\subseteq_P$*-IAR semantics for combined complexity.*

Proof.

Upper bounds. For the $\leq_w$-IAR semantics, we can use the following procedure to decide whether $\langle \mathcal{T}, \mathcal{A} \rangle \not\models_{\leq_w\text{-IAR}} q$:

1. Compute the weight u_{rep} of $\leq_w$-repairs (cf. proof of Proposition 3).
2. For every $\alpha \in \mathcal{A}$, use an NP oracle to decide if there exists a $\mathcal{T}$-consistent subset $\mathcal{B} \subseteq \mathcal{A}$ such that $\alpha \notin \mathcal{B}$ and $\sum_{\alpha \in \mathcal{B}} w(\alpha) = u_{\text{rep}}$. Let $\mathcal{A}'$ be the set of all assertions for which no such subset exists.
3. Use an NP oracle to verify that the CQ q is not entailed from $\langle \mathcal{T}, \mathcal{A}' \rangle$.

Correctness of the above procedure is straightforward: the ABox $\mathcal{A}'$ constructed in Step 2 is precisely the intersection of the $\leq_w$-repairs. The procedure runs in polynomial time with access to an NP oracle, yielding membership in Δ_2^p. For the bounded weight case, we first recall that the class $\Delta_2^p[O(\log n)]$ can be equivalently characterized as the class of decision problems which can be solved in polynomial-time with a single round of parallel calls to an NP oracle, cf. (Buss and Hay 1991). By using this technique, instead of computing u_{rep} in Step 1 by making a sequence of logarithmically many oracle calls, we can instead issue a single round of parallel calls to the NP oracle. Steps 2 and 3 can be implemented using two further rounds of parallel NP oracle calls. It follows from results in (Buss and Hay 1991) that we can reduce these three rounds into a single one, from which membership in $\Delta_2^p[O(\log n)]$ follows.

Similarly to the proof of Proposition 3, we can exploit Lemma 1 to obtain the upper bounds for the $\leq_P$-IAR and $\leq$-IAR semantics. For the $\subseteq_P$-IAR semantics, we can skip Step 1 of the procedure and modify Step 2 by using the polynomial-time procedure for identifying $\subseteq_P$-repairs from the proof of Proposition 3.

$\Delta_2^p[O(\log n)]$-hardness for CQs and $\subseteq_P$-IAR semantics. The proof is by reduction from the Parity(3SAT) problem introduced in the proof of Proposition 3. Let $\Phi = \varphi_1, \ldots, \varphi_n$ be a Parity(3SAT) instance satisfying the same restrictions as in the proof of Proposition 3, where each φ_i is a 3CNF over the variables $x_{i,1}, \ldots, x_{i,g_i}$ composed of m_i clauses $c_{i,1}, \ldots, c_{i,m_i}$. We combine ideas from the proof of Proposition 4 with a construction from (Bienvenu and Rosati 2013). In the latter paper, the authors define ABoxes $\mathcal{A}_j$ for the odd $j \in [1, n]$ and a Boolean CQ q with the following properties:

1. if $j_1 \neq j_2$, then $\mathsf{Inds}(\mathcal{A}_{j_1}) \cap \mathsf{Inds}(\mathcal{A}_{j_2}) = \emptyset$;
2. $\langle \emptyset, \bigcup_j \mathcal{A}_j \rangle \models q$ if and only if $\langle \emptyset, \mathcal{A}_j \rangle \models q$ for some j;
3. $\langle \emptyset, \mathcal{A}_j \rangle \models q$ if and only if φ_ℓ is satisfiable for $1 \leq \ell \leq j$;
4. there is a variable x in q and an individual a_j in each $\mathcal{A}_j$ such that $\langle \emptyset, \mathcal{A}_j \rangle \models q$ if and only if $\langle \emptyset, \mathcal{A}_j \rangle \models q[x \mapsto a_j]$.

The query q and ABoxes $\mathcal{A}_j$ provide a means of showing that the first j formulas are satisfiable, but we still also need a way of showing that the $j+1$st formula is unsatisfiable. To this end, we consider the TBox from the proof of Proposition 4:

$$\begin{aligned}\mathcal{T} = &\{\exists P^- \sqsubseteq \neg\exists N^-, \exists P \sqsubseteq \neg\exists U^-, \exists N \sqsubseteq \neg\exists U^-,\\ &\exists U \sqsubseteq \neg B, B \sqsubseteq \neg A\}\end{aligned}$$

where we assume that P, N, U, B, A are fresh roles and concepts, appearing neither in q nor any $\mathcal{A}_j$. We also create ABoxes $\mathcal{B}_j = \mathcal{B}_j^1 \cup \mathcal{B}_j^2 \cup \mathcal{B}_j^3$, for each odd $j \in [1, n]$:

$$\begin{aligned}\mathcal{B}_j^1 = &\{P(c_{j+1,\ell}, x_{j+1,h}) \mid x_{j+1,h} \in c_{j+1,\ell}\} \cup\\ &\{N(c_{j+1,\ell}, x_{j+1,h}) \mid \neg x_{j+1,h} \in c_{j+1,\ell}\}\\ \mathcal{B}_j^2 = &\{U(a_j, c_{j+1,\ell}) \mid 1 \leq \ell \leq m_{j+1}\}\\ \mathcal{B}_j^3 = &\{A(a_j), B(a_j)\}\end{aligned}$$

where the individuals of the forms $c_{j+1,\ell}$ and $x_{j+1,h}$ are fresh and do not appear in any $\mathcal{A}_j$ or any $\mathcal{B}_{j'}$ for $j' \neq j$, and the individual a_j is shared by $\mathcal{B}_j$ and $\mathcal{A}_j$. We then define the ABox $\mathcal{A}$ as the union of the $\mathcal{A}_j$ and $\mathcal{B}_j$, for all odd $j \in [1, n]$, and consider the following prioritization $\mathcal{P} = \langle \mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3 \rangle$:

$$\mathcal{P}_1 = \bigcup_{\mathsf{odd}\ j \in [1,n]} \mathcal{A}_j \cup \mathcal{B}_j^1$$

$$\mathcal{P}_\ell = \bigcup_{\text{odd } j \in [1,n]} \mathcal{B}_j^\ell \qquad \text{for } \ell \in \{2,3\}$$

We remark that since the signatures of the ABoxes $\mathcal{A}_j$ and $\mathcal{T}$ are disjoint, the assertions in the ABoxes $\mathcal{A}_j$ are not involved in any conflicts, and hence belong to every $\subseteq_P$ repair. Moreover, because the ABoxes $\mathcal{B}_j$ use disjoint sets of variables, the $\subseteq_P$-repairs of $\mathcal{A}$ w.r.t. $\mathcal{T}$ can be obtained by repairing each of the $\mathcal{B}_j$ separately (according to the prioritization $\mathcal{P}$) and taking the union of these repairs and the ABoxes $\mathcal{A}_j$. Using similar arguments to Proposition 4, one can show that $\langle \mathcal{T}, \mathcal{A}\rangle \models_{\subseteq_P\text{-IAR}} A(a_j)$ just in the case that φ_{j+1} is unsatisfiable.

Now let q' be the query obtained by adding $A(x)$ to q. We aim to show that $\langle \mathcal{T}, \mathcal{A}\rangle \models_{\subseteq_P\text{-IAR}} q'$ if and only if there is an odd $j \in [1, n]$ such that φ_j is satisfiable and φ_{j+1} is unsatisfiable.

First suppose that $\langle \mathcal{T}, \mathcal{A}\rangle \models_{\subseteq_P\text{-IAR}} q'$, and let $\mathcal{A}'$ be the intersection of the $\subseteq_P$-repairs. Then $\langle \mathcal{T}, \mathcal{A}'\rangle \models q'$, and since $\mathcal{T}$ contains only negative inclusions, we in fact have $\langle \emptyset, \mathcal{A}'\rangle \models q'$. Thus, there exists a function π mapping variables in q' to individuals in $\mathcal{A}'$ which witnesses the satisfaction of q'. It follows that $A(\pi(x)) \in \mathcal{A}'$, and hence $\pi(x) = a_\ell$ for some odd $\ell \in [1, n]$. By above, we know that this means that $\varphi_{\ell+1}$ is unsatisfiable. We have also seen above that $\mathcal{A}'$ contains all of the $\mathcal{A}_j$, and since q uses only predicates from the $\mathcal{A}_j$, we must have $\langle \emptyset, \bigcup_j \mathcal{A}_j\rangle \models q$. By Properties 2 and 4, we can infer that $\langle \emptyset, \mathcal{A}_\ell\rangle \models q$. Applying Property 3, we obtain that φ_ℓ is satisfiable.

For the other direction, suppose that φ_j is satisfiable and φ_{j+1} is unsatisfiable. Then by our earlier assumption, for every $1 \leq \ell \leq j$, the formula φ_ℓ is satisfiable. It follows then by Property 3 that $\langle \emptyset, \mathcal{A}_j\rangle \models q$, and so by Property 4, we must have $\langle \emptyset, \mathcal{A}_j\rangle \models q[x \mapsto a_j]$. Since $\mathcal{A}_j$ appears in all repairs, this yields $\langle \mathcal{T}, \mathcal{A}_j\rangle \models_{\subseteq_P\text{-IAR}} q[x \mapsto a_j]$. By earlier arguments, the unsatisfiability of φ_{j+1} means that $\langle \mathcal{T}, \mathcal{A}\rangle \models_{\subseteq_P\text{-IAR}} A(a_j)$ Putting this together, we obtain $\langle \mathcal{T}, \mathcal{A}\rangle \models_{\subseteq_P\text{-IAR}} q'$.

Remaining lower bounds. To show the Δ_2^p lower bounds, we can use the reduction from the proof of Proposition 3: as the TBox does not give any way of deriving $T(x_n)$, the query $T(x_n)$ holds in all $\leq_P$-repairs iff it holds in the intersection of all $\leq_P$-repairs. Finally, a $\Delta_2^p[O(\log n)]$ lower bound for AQ entailment under $\leq$-IAR semantics can be proved similarly to the corresponding result for the $\leq$-AR semantics (Proposition 3). □

11 Proofs for Section 5

Theorem 3 Let q be a Boolean CQ , $\mathcal{K} = \langle \mathcal{T}, \mathcal{A}\rangle$ be a DL-Lite$_\mathcal{R}$ KB, and $P = \langle \mathcal{P}_1, \ldots, \mathcal{P}_n\rangle$ be a prioritization of $\mathcal{A}$. Consider the following propositional formulas having variables of the form x_α for $\alpha \in \mathcal{A}$:

$$\varphi_{\neg q} = \bigwedge_{\mathcal{C} \in \mathsf{causes}(q)} (\bigvee_{\alpha \in \mathcal{C}} \bigvee_{\substack{\beta \in \mathsf{confl}(\alpha)\\ \beta \preceq_P \alpha}} x_\beta)$$

$$\varphi_{\mathsf{max}} = \bigwedge_{\alpha \in R_q} (x_\alpha \vee \bigvee_{\substack{\beta \in \mathsf{confl}(\alpha)\\ \beta \preceq_P \alpha}} x_\beta)$$

$$\varphi_{\mathsf{cons}} = \bigwedge_{\substack{\alpha,\beta \in R_q\\ \beta \in \mathsf{confl}(\alpha)}} \neg x_\alpha \vee \neg x_\beta$$

where $\mathsf{causes}(q)$ contains all causes for q in $\mathcal{K}$, $\mathsf{confl}(\alpha)$ contains all assertions β such that $\{\alpha, \beta\}$ is a conflict for $\mathcal{K}$, and R_q is the set of assertions reachable in $G_\mathcal{K}^P$ from some assertion β such that x_β appears in $\varphi_{\neg q}$. Then $\mathcal{K} \models_{\subseteq_P\text{-AR}} q$ iff $\varphi_{\neg q} \wedge \varphi_{\mathsf{max}} \wedge \varphi_{\mathsf{cons}}$ is unsatisfiable.

We begin by establishing the following lemmas that relate the $\subseteq_P$-repairs of $\mathcal{A}$ with the $\subseteq_P$-repairs of R_q.

Lemma 3. *Every $\subseteq_P$-repair of R_q can be extended to a $\subseteq_P$-repair of $\mathcal{A}$.*

Proof. Let $\mathcal{B}$ be a $\subseteq_P$-repair of R_q. Construct a set $\mathcal{A}'$ by adding to $\mathcal{B}$ a maximal subset $\mathcal{C}_1$ of assertions from $\mathcal{P}_1 \setminus \mathcal{B}$ such that $\mathcal{B} \cup \mathcal{C}_1$ is $\mathcal{T}$-consistent, then adding a maximal subset $\mathcal{C}_2$ of assertions from $\mathcal{P}_2 \setminus \mathcal{B}$ such that $\mathcal{B} \cup \mathcal{C}_1 \cup \mathcal{C}_2$ is $\mathcal{T}$-consistent, and so on. By construction, the set $\mathcal{A}'$ is $\mathcal{T}$-consistent, and we claim that it is in fact a $\subseteq_P$-repair of $\mathcal{A}$. Suppose for a contradiction that this is not the case. Then there must exist another $\mathcal{T}$-consistent set $\mathcal{D} \subseteq \mathcal{A}$ and some k such that $\mathcal{A}' \cap \mathcal{P}_i = \mathcal{D} \cap \mathcal{P}_i$ for every $1 \leq i < k$, and $\mathcal{A}' \cap \mathcal{P}_k \subsetneq \mathcal{D} \cap \mathcal{P}_k$. Consider some $\alpha \in (\mathcal{D} \cap \mathcal{P}_k) \setminus (\mathcal{A}' \cap \mathcal{P}_k)$. It follows from the construction of $\mathcal{A}'$ that $\mathcal{B} \cup (\mathcal{A}' \cap (\mathcal{P}_1 \cup \ldots \cup \mathcal{P}_k)) \cup \{\alpha\}$ is $\mathcal{T}$-inconsistent. Since $(\mathcal{A}' \cap (\mathcal{P}_1 \cup \ldots \cup \mathcal{P}_k)) \cup \{\alpha\}$ is a subset of $\mathcal{D}$, $\mathcal{D}$ is known to be $\mathcal{T}$-consistent, and all conflicts involve at most two assertions (Fact 1), it must be the case that α conflicts with some $\beta \in \mathcal{B} \setminus (\mathcal{P}_1 \cup \ldots \cup \mathcal{P}_k)$. Since $\beta \in \mathcal{B}$, it must belong to R_q. It follows then from the definition of R_q and the fact that $\alpha \in \mathcal{P}_k$ that the assertion α must also belong to R_q. Now consider the set $\mathcal{B}' = (\mathcal{B} \cap (\mathcal{P}_1 \cup \ldots \cup \mathcal{P}_k)) \cup \{\alpha\}$. It can be easily verified that $\mathcal{B}'$ is a $\mathcal{T}$-consistent set with $\mathcal{B} \subsetneq_P \mathcal{B}'$, contradicting our assumption that $\mathcal{B}$ is a $\subseteq_P$-repair of R_q. □

Lemma 4. *If $\mathcal{A}'$ is a $\subseteq_P$-repair of $\mathcal{A}$, then $\mathcal{A}' \cap R_q$ is a $\subseteq_P$-repair of R_q.*

Proof. Let $\mathcal{A}'$ be a $\subseteq_P$-repair of $\mathcal{A}$, and set $\mathcal{S} = \mathcal{A}' \cap R_q$. Clearly, $\mathcal{S}$ is $\mathcal{T}$-consistent. Suppose for a contradiction that there exists a set $\mathcal{S}' \subseteq R_q$ such that $\mathcal{S} \subsetneq_P \mathcal{S}'$, and let k be such that $\mathcal{S} \cap \mathcal{P}_i = \mathcal{S}_i' \cap \mathcal{P}_i$ for all $1 \leq i < k$ and $\mathcal{S} \cap \mathcal{P}_k \subsetneq \mathcal{S}' \cap \mathcal{P}_k$. We claim that

$$\mathcal{A}'' = ((\mathcal{A}' \setminus R_q) \cap (\mathcal{P}_1 \cup \ldots \cup \mathcal{P}_k)) \cup \mathcal{S}'$$

satisfies the following:

1. $\mathcal{A}''$ is $\mathcal{T}$-consistent.
2. $\mathcal{A}' \subsetneq_P \mathcal{A}''$.

Note that these statements together contradict our earlier assumption that $\mathcal{A}'$ is a $\subseteq_P$-repair.

To show the first statement, suppose for a contradiction that $\mathcal{A}''$ is $\mathcal{T}$-inconsistent. Since $\mathcal{A}' \setminus R_q$ and $\mathcal{S}'$ are both known to be $\mathcal{T}$-consistent, and conflicts in DL-Lite$_\mathcal{R}$ involve at most two assertions (Fact 1), there must exist a conflict $\{\alpha, \beta\}$ with $\alpha \in (\mathcal{A}' \setminus R_q) \cap (\mathcal{P}_1 \cup \ldots \cup \mathcal{P}_k)$ and $\beta \in \mathcal{S}'$. Moreover, since $\beta \in R_q$ and $\alpha \notin R_q$, we must have $\beta \prec_P \alpha$. The assertion α belongs to $\mathcal{P}_1 \cup \ldots \cup \mathcal{P}_k$, so we must have $\beta \in \mathcal{P}_j$ for some $j < k$. Since $\mathcal{S} \cap \mathcal{P}_i = \mathcal{S}' \cap \mathcal{P}_i$ for all

$1 \leq i < k$, it follows that $\beta \in \mathcal{S}$, hence $\{\alpha, \beta\} \subseteq \mathcal{A}'$. This is a contradiction, since $\mathcal{A}'$ was assumed to be a $\subseteq_P$-repair, and so must be $\mathcal{T}$-consistent.

For the second statement, we simply note that since $\mathcal{S} \cap \mathcal{P}_i = \mathcal{S}'_i \cap \mathcal{P}_i$ for all $1 \leq i < k$, we have $\mathcal{A}' \cap \mathcal{P}_i = \mathcal{A}'' \cap \mathcal{P}_i$ for every $1 \leq i < k$, and since $\mathcal{S} \cap \mathcal{P}_k \subsetneq \mathcal{S}' \cap \mathcal{P}_k$, we also have $\mathcal{A}' \cap \mathcal{P}_k \subsetneq \mathcal{A}'' \cap \mathcal{P}_k$. □

Proof of Theorem 3. We observe that the set of assertions α whose corresponding variable x_α appears in the formula $\varphi_{\neg q} \wedge \varphi_{\mathsf{max}} \wedge \varphi_{\mathsf{cons}}$ is precisely the set R_q. Moreover, every variable x_α with $\alpha \in R_q$ appears in the subformula φ_{max}.

For the first direction, suppose that the formula $\varphi_{\neg q} \wedge \varphi_{\mathsf{max}} \wedge \varphi_{\mathsf{cons}}$ is satisfiable, and let ν be a satisfying truth assignment. Consider the corresponding set of assertions $\mathcal{S}_\nu \subseteq R_q$ consisting of all those assertions α whose corresponding variable x_α is assigned to true by ν. As ν satisfies φ_{cons}, the set $\mathcal{S}_\nu$ contains no conflicts, i.e. it is $\mathcal{T}$-consistent. We claim that $\mathcal{S}_\nu$ is a $\subseteq_P$-repair of R_q. Suppose that this is not the case, and let $\mathcal{S}'$ be a $\mathcal{T}$-consistent subset of R_q such that $\mathcal{S} \cap \mathcal{P}_i = \mathcal{S}'_i \cap \mathcal{P}_i$ for all $1 \leq i < k$ and $\mathcal{S} \cap \mathcal{P}_k \subsetneq \mathcal{S}' \cap \mathcal{P}_k$. Consider some $\alpha \in (\mathcal{S}' \setminus \mathcal{S}) \cap \mathcal{P}_k$. Since $\alpha \notin \mathcal{S}$, we must have $\nu(x_\alpha) = \mathsf{false}$. As φ_{max} is satisfied by ν, there must exist some variable x_β with $\nu(x_\beta) = \mathsf{true}$ such that the corresponding assertion β satisfies $\beta \in \mathsf{confl}(\alpha)$ and $\beta \preceq_P \alpha$. However, we know that $\mathcal{S} \cap \mathcal{P}_i \subseteq \mathcal{S}'_i \cap \mathcal{P}_i$ for every $1 \leq i \leq k$, hence $\beta \in \mathcal{S}'$, contradicting the supposed consistency of $\mathcal{S}'$. We have thus shown that $\mathcal{S}_\nu$ is a $\subseteq_P$-repair of R_q. Applying Lemma 3, we can find a $\subseteq_P$-repair $\mathcal{A}'$ of $\mathcal{A}$ such that $\mathcal{S}_\nu \subseteq \mathcal{A}'$. To show that $\langle \mathcal{T}, \mathcal{A}' \rangle \not\models q$, consider some cause $\mathcal{C}$ for q in $\mathcal{K}$. Then since ν satisfies $\varphi_{\neg q}$, there must exist some assertion $\alpha \in \mathcal{C}$ and some $\beta \in \mathsf{confl}(\alpha)$ such that $\nu(x_\beta) = \mathsf{true}$. It follows that $\beta \in \mathcal{S}$, hence $\beta \in \mathcal{A}'$ and $\mathcal{C} \not\subseteq \mathcal{A}'$. We have thus showed that $\mathcal{A}'$ contains no cause for q. We can thus conclude that $\mathcal{K} \not\models_{\subseteq_P\text{-AR}} q$.

For the other direction, suppose that $\mathcal{K} \not\models_{\subseteq_P\text{-AR}} q$, and let $\mathcal{S}$ be a $\subseteq_P$-repair of $\mathcal{A}$ such that $\langle \mathcal{T}, \mathcal{S} \rangle \not\models q$. Consider the set $\mathcal{S}' = \mathcal{S} \cap R_q$. By Lemma 4, we have that $\mathcal{S}'$ is a $\subseteq_P$-repair of R_q. Let $\nu_{\mathcal{S}'}$ be the truth assignment that assigns to true precisely those variables x_α for which $\alpha \in \mathcal{S}'$. We wish to show that $\nu_{\mathcal{S}'}$ satisfies $\varphi_{\neg q} \wedge \varphi_{\mathsf{max}} \wedge \varphi_{\mathsf{cons}}$. First, consider some cause $\mathcal{C} \subseteq \mathcal{A}$ for q w.r.t. $\mathcal{K}$. Since $\langle \mathcal{T}, \mathcal{S} \rangle \not\models q$, there must exist an assertion $\alpha \in \mathcal{C}$ that does not appear in $\mathcal{S}$. We also know that $\mathcal{S}$ is a $\subseteq_P$-repair, so there must exist some $\beta \in \mathcal{S}$ with $\beta \preceq_P \alpha$ that conflicts with α (otherwise, we could obtain a more preferred subset by adding α to $\mathcal{S}$ and removing any assertions conflicting with α). The assertion β belongs to R_q, so the variable x_β will be assigned to true by $\nu_{\mathcal{S}'}$, and the clause in $\varphi_{\neg q}$ that corresponds to cause $\mathcal{C}$ is satisfied by $\nu_{\mathcal{S}'}$. We have thus shown that every clause in $\varphi_{\neg q}$ is satisfied.

Next, consider an assertion $\alpha \in R_q$ and its associated clause $x_\alpha \vee \bigvee_{\beta \in \mathsf{confl}(\alpha), \beta \preceq_P \alpha} x_\beta$ in the formula φ_{max}. If $\alpha \in \mathcal{S}'$, then x_α will be assigned true by $\nu_{\mathcal{S}'}$, and the clause is satisfied. If instead $\alpha \notin \mathcal{S}'$, then also $\alpha \notin \mathcal{S}$. Using the fact that $\mathcal{S}$ is a $\subseteq_P$-repair of $\mathcal{A}$ and similar arguments to above, we can infer that there is some there must exist some $\beta \in \mathcal{S}$ with $\beta \preceq_P \alpha$ and $\beta \in \mathsf{confl}(\alpha)$. Since $\alpha \in R_q$, it follows from the definition of the set R_q that $\beta \in R_q$, hence $\beta \in \mathcal{S}'$. We thus have $\nu_{\mathcal{S}'}(x_\beta) = \mathsf{true}$, and so the clause for α is satisfied. This proves that φ_{max} is satisfied by $\nu_{\mathcal{S}'}$.

Finally, since $\mathcal{S}'$ is $\mathcal{T}$-consistent, it contains no conflicts, and so $\nu_{\mathcal{S}'}$ satisfies φ_{cons}. We have thus exhibited a satisfying assignment for the formula $\varphi_{\neg q} \wedge \varphi_{\mathsf{max}} \wedge \varphi_{\mathsf{cons}}$. □

Theorem 4 Let q, $\mathcal{K}$, P, $\mathsf{causes}(q)$, and $\mathsf{confl}(\alpha)$ be as in Theorem 3. For each $\mathcal{C} \in \mathsf{causes}(q)$, consider the formulas:

$$\varphi_{\neg\mathcal{C}} = \bigvee_{\alpha \in \mathcal{C}} \bigvee_{\substack{\beta \in \mathsf{confl}(\alpha) \\ \beta \preceq_P \alpha}} x_\beta$$

$$\varphi^{\mathcal{C}}_{\mathsf{max}} = \bigwedge_{\alpha \in R_{\mathcal{C}}} (x_\alpha \vee \bigvee_{\substack{\beta \in \mathsf{confl}(\alpha) \\ \beta \preceq_P \alpha}} x_\beta)$$

$$\varphi^{\mathcal{C}}_{\mathsf{cons}} = \bigwedge_{\substack{\alpha, \beta \in R_{\mathcal{C}} \\ \beta \in \mathsf{confl}(\alpha)}} \neg x_\alpha \vee \neg x_\beta$$

where $R_{\mathcal{C}}$ is the set of assertions reachable in $G^P_{\mathcal{K}}$ from some assertion β such that x_β appears in $\varphi_{\neg\mathcal{C}}$. Then $\mathcal{K} \models_{\subseteq_P\text{-IAR}} q$ iff there exists $\mathcal{C} \in \mathsf{causes}(q)$ such that the formula $\varphi_{\neg\mathcal{C}} \wedge \varphi^{\mathcal{C}}_{\mathsf{cons}} \wedge \varphi^{\mathcal{C}}_{\mathsf{max}}$ is unsatisfiable.

Proof. Using similar arguments to the proof of Theorem 3, one can show that $\varphi_{\neg\mathcal{C}} \wedge \varphi^{\mathcal{C}}_{\mathsf{cons}} \wedge \varphi^{\mathcal{C}}_{\mathsf{max}}$ is satisfiable if and only if there exists a $\subseteq_P$-repair of $\mathcal{A}$ which does not contain the cause $\mathcal{C}$. It follows that $\varphi_{\neg\mathcal{C}} \wedge \varphi^{\mathcal{C}}_{\mathsf{cons}} \wedge \varphi^{\mathcal{C}}_{\mathsf{max}}$ is satisfiable for every $\mathcal{C} \in \mathsf{causes}(q)$ just in the case that there is no cause of q in the intersection of the $\subseteq_P$-repairs of $\mathcal{A}$ w.r.t. $\mathcal{T}$, i.e. $\mathcal{K} \not\models_{\subseteq_P\text{-IAR}} q$. □

12 Datasets

Our ABoxes were generated with EUDG by setting its *data completeness* parameter (i.e., the percentage of individuals from a given concept for which roles describing this concept are indeed filled) to 95%, which seems realistic from the application viewpoint. All the generated ABoxes were found consistent w.r.t. our enriched TBox, meaning that the added disjointness constraints were to some extent faithful to the reused benchmark.

Generation of Data Inconsistencies Inconsistencies were introduced by contradicting the presence of an individual in a concept assertion with probability p, and the presence of each individual in a role assertion with probability $p/2$. A contradicting assertion is built by stating that the considered individual also belongs to a disjoint but close concept, i.e., the two concepts have a common super-concept which is not the top concept "Thing". Note that a concept may here be an unqualified existential role restriction. The contradicting assertion is added either explicitly or implicitly by choosing one of its specializations (obtained by query rewriting). We chose to generate such inconsistencies because they seem quite natural in real applications (e.g., using by mistake AssistantProfessor in place of AssociateProfessor). Conflicting assertions thus introduced are in turn processed as described above. Additionally, for every role assertion, its individuals are switched with probability $p/10$. We chose to generate such misuses of roles because they seem quite natural mistakes in real applications, that may lead to inconsistencies (e.g., inverting the Faculty and Course in a teacherOf role assertion).

Prioritization Prioritizations of an inconsistent ABox were made either by choosing the *same* priority level for *all* the assertions of a concept/role or by choosing *a* priority level for *each* ABox assertion. Moreover, priority level choices were considered either equiprobable or not. We built prioritizations this way to capture a variety of scenarios. For instance, a database administrator may (manually) partition an ABox using a few priority levels set by concept/role, based upon the reliability of the business processes that provide the data; an ABox integrating data from many sources may be partitioned with more priority levels set per assertion, with the priority of an assertion depending on the reliability of the sources from which it originates.

13 Experimental results

Our experiments ran on an Intel i5-3470 CPU server at 3.20 GHz, with 8 Go of RAM, and running Windows 7. All reported times are averaged over 10 runs.

The time it took CQAPri to be up and ready to answer queries is dominated by the construction of the oriented conflict graph for the ABox (Table 1); it took only a few tens of milliseconds to load the TBox and open the PostgreSQL connection to the ABox.

Table 3 shows, per ABox and query pair, how many IAR answers and AR answers (not already found to be IAR) were identified among the possible ones, as well as the number of possible answers that are neither IAR nor AR; it also gives AR query answering time. In this table, OOM means that CQAPri ran out of memory.

Most of the queries (4 out of 5) with AR non-IAR answers have one or two atoms: `g2`, `g3`, `q1`, and `q2`. Such answers show up as these queries are general and involve concepts with many disjoint sub-concepts. `Lutz5` also provides a few AR non-IAR answers in some rare cases (2 out of 12, with ∼60% of conflicting assertions), while it is much more complex.

Table 4 shows, per query and for some prioritized ABoxes, how many answers were in the approximation of $\subseteq_P$-IAR, the number of $\subseteq_P$-IAR answers (not already found to be in the approximation), of $\subseteq_P$-AR answers (not already found to be $\subseteq_P$-IAR) and the number of possible answers that are not consistent; it also gives $\subseteq_P$-AR query answering time. Figure 3 shows the time spent by CQAPri for AR and $\subseteq_P$-AR query answering on four sets of ABoxes when the proportion of conflicting assertions is a few percent, as it is likely to be in most real applications, and for three different prioritizations. It shows that our approach scales for these sets of ABoxes.

We observed that adding prioritizations to the ABoxes complicates query answering, and using 3 priority levels typically led to harder instances than using 10 levels. Moreover, three-level prioritizations set per assertion were more difficult than three levels set per concept/role. By contrast, choosing to make priority levels equiprobable or not had little effect, except when some levels almost disappear due to heavily unbalanced probabilities.

14 Queries and Negative Inclusions

Figure 4 contains the queries utilized in our experiments, and below we give the complete list of the 875 negative inclusions added to the $\text{LUBM}^{\exists}_{20}$ TBox (comprising 127 concepts, 27 roles and 152 positive inclusions) . This apparently huge number of added constraints results from the many pairwise disjoint concepts/roles used in the TBox. These inclusions state the disjointness of pairs of concepts or roles appearing at the same level in the TBox (that is, having the same distance to the top concept "Thing") and having the same closest super-concept. We excluded a small number of such inclusions when they did not seem to reflect the intended meaning of the concepts / roles.

```
DisjointClasses(:AdministrativeStaff :Faculty)
DisjointClasses(:Article :Book)
DisjointClasses(:Article :Manual)
DisjointClasses(:Article :Software)
DisjointClasses(:Article :Specification)
DisjointClasses(:Article :UnofficialPublication)
DisjointClasses(:AssistantProfessor :AssociateProfessor)
DisjointClasses(:AssistantProfessor :Dean)
DisjointClasses(:AssistantProfessor :ExDean)
DisjointClasses(:AssistantProfessor :FullProfessor)
DisjointClasses(:AssistantProfessor :VisitingProfessor)
DisjointClasses(:AssociateProfessor :AssistantProfessor)
DisjointClasses(:AssociateProfessor :Dean)
DisjointClasses(:AssociateProfessor :ExDean)
DisjointClasses(:AssociateProfessor :FullProfessor)
DisjointClasses(:AssociateProfessor :VisitingProfessor)
DisjointClasses(:Book :Article)
DisjointClasses(:Book :Manual)
DisjointClasses(:Book :Software)
DisjointClasses(:Book :Specification)
DisjointClasses(:Book :UnofficialPublication)
DisjointClasses(:ClericalStaff :SystemsStaff)
DisjointClasses(:College :Department)
DisjointClasses(:College :Institute)
DisjointClasses(:College :Program)
DisjointClasses(:College :ResearchGroup)
DisjointClasses(:College :University)
DisjointClasses(:ConferencePaper :JournalArticle)
DisjointClasses(:ConferencePaper :TechnicalReport)
DisjointClasses(:Course :Exam)
DisjointClasses(:Course :ExamRecord)
DisjointClasses(:Course :Research)
DisjointClasses(:Dean :AssistantProfessor)
DisjointClasses(:Dean :AssociateProfessor)
DisjointClasses(:Dean :ExDean)
DisjointClasses(:Dean :FullProfessor)
DisjointClasses(:Dean :VisitingProfessor)
DisjointClasses(:Department :College)
DisjointClasses(:Department :Institute)
DisjointClasses(:Department :Program)
DisjointClasses(:Department :ResearchGroup)
DisjointClasses(:Department :University)
DisjointClasses(:Director :GraduateStudent)
DisjointClasses(:Director :ResearchAssistant)
DisjointClasses(:Director :Student)
DisjointClasses(:Director :TeachingAssistant)
DisjointClasses(:Employee :TeachingAssistant)
DisjointClasses(:ExDean :AssistantProfessor)
DisjointClasses(:ExDean :AssociateProfessor)
DisjointClasses(:ExDean :Dean)
DisjointClasses(:ExDean :FullProfessor)
DisjointClasses(:ExDean :VisitingProfessor)
DisjointClasses(:Exam :Course)
DisjointClasses(:Exam :ExamRecord)
DisjointClasses(:Exam :Research)
DisjointClasses(:ExamRecord :Course)
DisjointClasses(:ExamRecord :Exam)
DisjointClasses(:ExamRecord :Research)
DisjointClasses(:Faculty :AdministrativeStaff)
DisjointClasses(:FullProfessor :AssistantProfessor)
DisjointClasses(:FullProfessor :AssociateProfessor)
DisjointClasses(:FullProfessor :Dean)
DisjointClasses(:FullProfessor :ExDean)
DisjointClasses(:FullProfessor :VisitingProfessor)
```

<table>
<tr><th>ABox / query</th><th>req2</th><th>req3</th><th>g2</th><th>g3</th><th>q1</th><th>q2</th><th>q4</th><th>Lutz1</th><th>Lutz5</th></tr>
<tr><td>u1p15e-4</td><td>1519|0|25
44ms</td><td>231|0|4
163ms</td><td>1184|9|6
64ms</td><td>1066|2|7
57ms</td><td>20119|0|284
450ms</td><td>8309|10|60
277ms</td><td>79907|0|3818
1080ms</td><td>15081|0|2443
17175ms</td><td>3136|0|37
2316ms</td></tr>
<tr><td>u1p5e-2</td><td>883|0|663
72ms</td><td>85|0|144
167ms</td><td>946|221|164
273ms</td><td>908|101|189
74ms</td><td>11430|5|8870
1077ms</td><td>6520|634|1440
444ms</td><td>13520|0|71044
4124ms</td><td>495|0|16713
19828ms</td><td>2010|0|1152
2610ms</td></tr>
<tr><td>u1p2e-1</td><td>243|0|1303
106ms</td><td>5|0|213
191ms</td><td>546|568|615
2084ms</td><td>534|283|690
237ms</td><td>2894|40|17100
3691ms</td><td>3028|1410|4808
1478ms</td><td>292|0|77267
5146ms</td><td>0|0|16348
23677ms</td><td>543|5|2534
3049ms</td></tr>
<tr><td>u5p15e-4</td><td>10082|0|165
62ms</td><td>1403|0|34
831ms</td><td>2531|52|29
215ms</td><td>7341|14|27
119ms</td><td>133525|0|2278
3870ms</td><td>53319|51|318
2001ms</td><td>460955|0|70115
8627ms</td><td>105389|0|11157
33704ms</td><td>20333|0|245
21766ms</td></tr>
<tr><td>u5p5e-2</td><td>6014|0|4239
285ms</td><td>591|0|822
942ms</td><td>1643|750|1143
1029ms</td><td>6340|608|1195
297ms</td><td>77289|76|57786
9598ms</td><td>43270|3490|8538
3595ms</td><td>7844|0|516488
32099ms</td><td>2104|0|112321
52586ms</td><td>13191|0|7311
24897ms</td></tr>
<tr><td>u5p2e-1</td><td>1592|0|8688
474ms</td><td>63|0|1287
1182ms</td><td>1189|1024|4093
5658ms</td><td>3895|1964|4636
1239ms</td><td>18780|376|114015
30928ms</td><td>20925|10726|28456
12189ms</td><td>0|0|506361
48790ms</td><td>0|0|107905
83980ms</td><td>3656|0|16565
29606ms</td></tr>
<tr><td>u10p15e-4</td><td>19008|0|305
88ms</td><td>2687|0|82
1650ms</td><td>3893|90|92
352ms</td><td>13513|24|87
182ms</td><td>251991|0|4005
7478ms</td><td>102634|113|577
3639ms</td><td>769786|0|214146
18729ms</td><td>189519|0|30275
51906ms</td><td>38244|0|509
40615ms</td></tr>
<tr><td>u10p5e-2</td><td>11334|0|7993
453ms</td><td>1179|0|1546
1786ms</td><td>2744|908|2147
1881ms</td><td>11669|1064|2327
562ms</td><td>145488|119|109193
18233ms</td><td>82568|6048|17844
7471ms</td><td>1001|0|976391
107063ms</td><td>5249|0|210086
85541ms</td><td>25163|0|13391
45994ms</td></tr>
<tr><td>u10p2e-1</td><td>2997|0|16373
891ms</td><td>105|0|2503
2408ms</td><td>2269|1075|7702
10796ms</td><td>6870|3734| 8629
2591ms</td><td>35086|727|215197
68364ms</td><td>37948|20613|56376
32523ms</td><td>OOM</td><td>0|0|202507
175905ms</td><td>6797|3|31179
56572ms</td></tr>
<tr><td>u20p15e-4</td><td>40428|0|714
170ms</td><td>5569|0|159
5286ms</td><td>7159|161|186
739ms</td><td>28889|62|175
401ms</td><td>535341|0|9686
19453ms</td><td>210179|191|1158
9044ms</td><td>OOM</td><td>414600|0|53566
112056ms</td><td>81996|0|1094
100479ms</td></tr>
<tr><td>u20p5e-2</td><td>24183|0|16978
1417ms</td><td>2385|0|3259
6507ms</td><td>5727|937|4558
4970ms</td><td>24976|2293|4872
1048ms</td><td>309625|259|232521
45615ms</td><td>170339|13003|34625
21271ms</td><td>OOM</td><td>15539|0|443696
203818ms</td><td>53420|0|29235
110658ms</td></tr>
<tr><td>u20p2e-1</td><td>6336|0|34964
5695ms</td><td>203|0|5185
7468ms</td><td>4803|1550|16206
74618ms</td><td>14845|7894|18394
13571ms</td><td>OOM</td><td>OOM</td><td>OOM</td><td>OOM</td><td>14355|0|67071
495490ms</td></tr>
</table>

Table 3: Number of IAR|AR non-IAR|possible non-AR answers and query answering time under the (plain) AR semantics, per ABox and query

<table>
<tr><th>ABox / query</th><th>req2</th><th>req3</th><th>g2</th><th>g3</th><th>q1</th><th>q2</th><th>q4</th><th>Lutz1</th><th>Lutz5</th></tr>
<tr><td>u1p15e-4l3cr=</td><td>1527|0
0|17
44ms</td><td>231|1
0|3
157ms</td><td>1190|0
3|6
68ms</td><td>1068|1
1|5
57ms</td><td>20153|32
0|218
471ms</td><td>8314|21
5|39
286ms</td><td>81215|390
0|2120
1258ms</td><td>15110|26
0|2388
16487ms</td><td>3144|13
0|16
2316ms</td></tr>
<tr><td>u1p5e-2l3cr=</td><td>981|292
0|273
72ms</td><td>100|46
0|83
185ms</td><td>1136|1
46|148
168ms</td><td>989|40
16|153
93ms</td><td>13416|4813
0|2076
8629ms</td><td>7269|177
45|1103
444ms</td><td>24707|15724
0|44133
13523ms</td><td>565|13142
0|3501
37030ms</td><td>2281|22
0|859
2603ms</td></tr>
<tr><td>u10p15e-4l3cr=</td><td>1908|170
0|135
97ms</td><td>2687|40
0|42
1740ms</td><td>4026|0
15|34
323ms</td><td>13541|23
7|53
199ms</td><td>252136|2050
0|1810
8305ms</td><td>102735|343
24|222
4170ms</td><td>835886|100766
0|47280
91801ms</td><td>189607|25700
0|4487
61148ms</td><td>38339|157
0|257
43706ms</td></tr>
<tr><td>u10p5e-2l3cr=</td><td>14762|0
0|4565
347ms</td><td>1921|20
0|784
1878ms</td><td>3265|1
733|1800
4388ms</td><td>13388|401
217|1054
1500ms</td><td>172713|21
0|82066
327806ms</td><td>91269|9971
559|4661
38518ms</td><td>4509|28128
0|944755
1751616ms</td><td>6720|131105
0|77510
1210720ms</td><td>32924|3786
0|1844
61792ms</td></tr>
<tr><td>u1p15e-4l10a=</td><td>1531|11
0|2
41ms</td><td>231|4
0|0
171ms</td><td>1194|0
1|4
62ms</td><td>1071|1
0|3
62ms</td><td>20234|123
0|46
425ms</td><td>8332|34
0|13
287ms</td><td>80597|2132
0|996
983ms</td><td>15860|1617
0|47
15765ms</td><td>3161|8
0|4
2375ms</td></tr>
<tr><td>u1p5e-2l10a=</td><td>1126|302
0|118
67ms</td><td>116|85
10|28
176ms</td><td>1172|3
27|129
113ms</td><td>1062|31
6|99
68ms</td><td>15068|4396
0|841
1903ms</td><td>7589|665
2|328
409ms</td><td>26355|36461
0|21748
13005ms</td><td>2295|13506
0|1407
23822ms</td><td>2741|231
0|190
2636ms</td></tr>
<tr><td>u10p15e-4l10a=</td><td>19152|133
0|28
83ms</td><td>2721|40
0|8
1717ms</td><td>4000|1
11|63
312ms</td><td>13567|12
1|44
207ms</td><td>253892|1831
0|273
7865ms</td><td>102977|223
4|120
4239ms</td><td>859425|97591
0|26916
53977ms</td><td>198019|21454
0|321
52791ms</td><td>38606|81
0|66
44322ms</td></tr>
<tr><td>u10p5e-2l10a=</td><td>14403|3755
0|1169
431ms</td><td>1537|956
0|232
1856ms</td><td>3840|6
163|1790
1229ms</td><td>13514|352
69|1125
435ms</td><td>189643|53091
0|12066
48585ms</td><td>95153|6748
202|4357
6834ms</td><td>Time out

(1h)</td><td>19009|144184
0|52142
441268ms</td><td>33524|2746
0|2284
49042ms</td></tr>
</table>

Table 4: Number of answers that are in the approximation of $\subseteq_P$-IAR $|\subseteq_P$-IAR but not in the approximation of $\subseteq_P$-IAR $|\subseteq_P$-AR non-$\subseteq_P$-IAR |possible non-$\subseteq_P$-AR, and query answering time under $\subseteq_P$-AR, per ABox and query

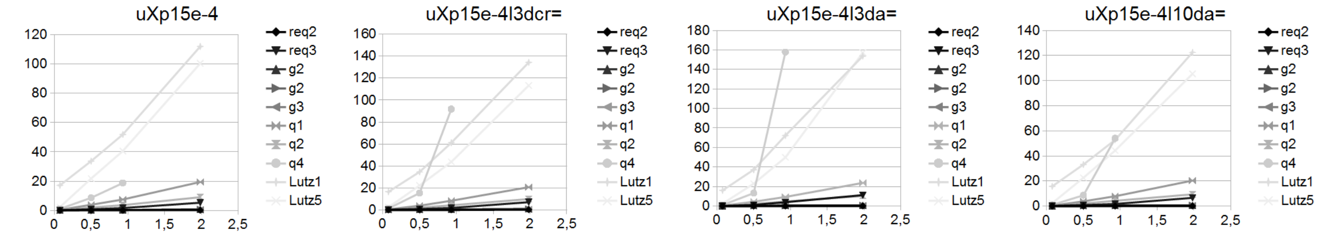

Figure 3: Time (in sec.) spent by CQAPri for AR and $\subseteq_P$-AR query answering on four sets of ABoxes (in millions of assertions)

$$
\begin{aligned}
req2(x,y) &\leftarrow \mathsf{Person}(x), \mathsf{teacherOf}(x,y), \mathsf{Course}(y)\\
req3(x,y,z) &\leftarrow \mathsf{Student}(x), \mathsf{advisor}(x,y), \mathsf{Faculty}(y), \mathsf{takesCourse}(x,z), \mathsf{teacherOf}(y,z), \mathsf{Course}(z)\\
g2(x) &\leftarrow \mathsf{Organization}(x)\\
g3(x) &\leftarrow \mathsf{Employee}(x)\\
q1(x,y) &\leftarrow \mathsf{Person}(x), \mathsf{takesCourse}(x,y)\\
q2(x,y) &\leftarrow \mathsf{Employee}(x), \mathsf{publicationAuthor}(y,x)\\
q4(x,y,z,u,v,w) &\leftarrow \mathsf{FullProfessor}(x), \mathsf{publicationAuthor}(y,x), \mathsf{teacherOf}(x,z), \mathsf{advisor}(u,x),\\
&\qquad \mathsf{GraduateStudent}(u), \mathsf{degreeFrom}(x,v), \mathsf{degreeFrom}(u,w)\\
Lutz1(x,y) &\leftarrow \mathsf{Student}(x), \mathsf{takesCourse}(x,y), \mathsf{Course}(y), \mathsf{teacherOf}(z,y), \mathsf{Faculty}(z), \mathsf{worksFor}(z,u),\\
&\qquad \mathsf{Department}(u), \mathsf{memberOf}(x,u)\\
Lutz5(x) &\leftarrow \mathsf{Publication}(x), \mathsf{publicationAuthor}(x,y), \mathsf{Professor}(y), \mathsf{publicationAuthor}(x,z), \mathsf{Student}(z)
\end{aligned}
$$

Figure 4: Queries

```
DisjointClasses(:GraduateStudent :Director)
DisjointClasses(:GraduateStudent :chair)
DisjointClasses(:Institute :College)
DisjointClasses(:Institute :Department)
DisjointClasses(:Institute :Program)
DisjointClasses(:Institute :ResearchGroup)
DisjointClasses(:Institute :University)
DisjointClasses(:JournalArticle :ConferencePaper)
DisjointClasses(:JournalArticle :TechnicalReport)
DisjointClasses(:Lecturer :PostDoc)
DisjointClasses(:Lecturer :Professor)
DisjointClasses(:Manual :Article)
DisjointClasses(:Manual :Book)
DisjointClasses(:Manual :Software)
DisjointClasses(:Manual :Specification)
DisjointClasses(:Manual :UnofficialPublication)
DisjointClasses(:Organization :Person)
DisjointClasses(:Organization :Publication)
DisjointClasses(:Organization :Schedule)
DisjointClasses(:Organization :Work)
DisjointClasses(:Person :Organization)
DisjointClasses(:Person :Publication)
DisjointClasses(:Person :Schedule)
DisjointClasses(:Person :Work)
DisjointClasses(:PostDoc :Lecturer)
DisjointClasses(:PostDoc :Professor)
DisjointClasses(:Professor :Lecturer)
DisjointClasses(:Professor :PostDoc)
DisjointClasses(:Program :College)
DisjointClasses(:Program :Department)
DisjointClasses(:Program :Institute)
DisjointClasses(:Program :ResearchGroup)
DisjointClasses(:Program :University)
DisjointClasses(:Publication :Organization)
DisjointClasses(:Publication :Person)
DisjointClasses(:Publication :Schedule)
DisjointClasses(:Publication :Work)
DisjointClasses(:Research :Course)
DisjointClasses(:Research :Exam)
DisjointClasses(:Research :ExamRecord)
DisjointClasses(:ResearchAssistant :Director)
DisjointClasses(:ResearchAssistant :chair)
DisjointClasses(:ResearchGroup :College)
DisjointClasses(:ResearchGroup :Department)
DisjointClasses(:ResearchGroup :Institute)
DisjointClasses(:ResearchGroup :Program)
DisjointClasses(:ResearchGroup :University)
DisjointClasses(:Schedule :Organization)
DisjointClasses(:Schedule :Person)
DisjointClasses(:Schedule :Publication)
DisjointClasses(:Schedule :Work)
DisjointClasses(:Software :Article)
DisjointClasses(:Software :Book)
DisjointClasses(:Software :Manual)
DisjointClasses(:Software :Specification)
DisjointClasses(:Software :UnofficialPublication)
DisjointClasses(:Specification :Article)
DisjointClasses(:Specification :Book)
DisjointClasses(:Specification :Manual)
DisjointClasses(:Specification :Software)
DisjointClasses(:Specification :UnofficialPublication)
DisjointClasses(:Student :Director)
DisjointClasses(:Student :chair)
DisjointClasses(:Subj10Course :Subj11Course)
DisjointClasses(:Subj10Course :Subj12Course)
DisjointClasses(:Subj10Course :Subj13Course)
DisjointClasses(:Subj10Course :Subj14Course)
DisjointClasses(:Subj10Course :Subj15Course)
DisjointClasses(:Subj10Course :Subj16Course)
DisjointClasses(:Subj10Course :Subj17Course)
DisjointClasses(:Subj10Course :Subj18Course)
DisjointClasses(:Subj10Course :Subj19Course)
DisjointClasses(:Subj10Course :Subj1Course)
DisjointClasses(:Subj10Course :Subj20Course)
DisjointClasses(:Subj10Course :Subj2Course)
DisjointClasses(:Subj10Course :Subj3Course)
DisjointClasses(:Subj10Course :Subj4Course)
DisjointClasses(:Subj10Course :Subj5Course)
DisjointClasses(:Subj10Course :Subj6Course)
DisjointClasses(:Subj10Course :Subj7Course)
DisjointClasses(:Subj10Course :Subj8Course)
DisjointClasses(:Subj10Course :Subj9Course)
DisjointClasses(:Subj10Department :Subj11Department)
DisjointClasses(:Subj10Department :Subj12Department)
DisjointClasses(:Subj10Department :Subj13Department)
DisjointClasses(:Subj10Department :Subj14Department)
DisjointClasses(:Subj10Department :Subj15Department)
DisjointClasses(:Subj10Department :Subj16Department)
DisjointClasses(:Subj10Department :Subj17Department)
DisjointClasses(:Subj10Department :Subj18Department)
DisjointClasses(:Subj10Department :Subj19Department)
DisjointClasses(:Subj10Department :Subj1Department)
DisjointClasses(:Subj10Department :Subj20Department)
DisjointClasses(:Subj10Department :Subj2Department)
DisjointClasses(:Subj10Department :Subj3Department)
DisjointClasses(:Subj10Department :Subj4Department)
DisjointClasses(:Subj10Department :Subj5Department)
DisjointClasses(:Subj10Department :Subj6Department)
DisjointClasses(:Subj10Department :Subj7Department)
DisjointClasses(:Subj10Department :Subj8Department)
```

```
DisjointClasses(:Subj10Department :Subj9Department)
DisjointClasses(:Subj10Professor :Subj11Professor)
DisjointClasses(:Subj10Professor :Subj12Professor)
DisjointClasses(:Subj10Professor :Subj13Professor)
DisjointClasses(:Subj10Professor :Subj14Professor)
DisjointClasses(:Subj10Professor :Subj15Professor)
DisjointClasses(:Subj10Professor :Subj16Professor)
DisjointClasses(:Subj10Professor :Subj17Professor)
DisjointClasses(:Subj10Professor :Subj18Professor)
DisjointClasses(:Subj10Professor :Subj19Professor)
DisjointClasses(:Subj10Professor :Subj1Professor)
DisjointClasses(:Subj10Professor :Subj20Professor)
DisjointClasses(:Subj10Professor :Subj2Professor)
DisjointClasses(:Subj10Professor :Subj3Professor)
DisjointClasses(:Subj10Professor :Subj4Professor)
DisjointClasses(:Subj10Professor :Subj5Professor)
DisjointClasses(:Subj10Professor :Subj6Professor)
DisjointClasses(:Subj10Professor :Subj7Professor)
DisjointClasses(:Subj10Professor :Subj8Professor)
DisjointClasses(:Subj10Professor :Subj9Professor)
DisjointClasses(:Subj11Course :Subj10Course)
DisjointClasses(:Subj11Course :Subj12Course)
DisjointClasses(:Subj11Course :Subj13Course)
DisjointClasses(:Subj11Course :Subj14Course)
DisjointClasses(:Subj11Course :Subj15Course)
DisjointClasses(:Subj11Course :Subj16Course)
DisjointClasses(:Subj11Course :Subj17Course)
DisjointClasses(:Subj11Course :Subj18Course)
DisjointClasses(:Subj11Course :Subj19Course)
DisjointClasses(:Subj11Course :Subj1Course)
DisjointClasses(:Subj11Course :Subj20Course)
DisjointClasses(:Subj11Course :Subj2Course)
DisjointClasses(:Subj11Course :Subj3Course)
DisjointClasses(:Subj11Course :Subj4Course)
DisjointClasses(:Subj11Course :Subj5Course)
DisjointClasses(:Subj11Course :Subj6Course)
DisjointClasses(:Subj11Course :Subj7Course)
DisjointClasses(:Subj11Course :Subj8Course)
DisjointClasses(:Subj11Course :Subj9Course)
DisjointClasses(:Subj11Department :Subj10Department)
DisjointClasses(:Subj11Department :Subj12Department)
DisjointClasses(:Subj11Department :Subj13Department)
DisjointClasses(:Subj11Department :Subj14Department)
DisjointClasses(:Subj11Department :Subj15Department)
DisjointClasses(:Subj11Department :Subj16Department)
DisjointClasses(:Subj11Department :Subj17Department)
DisjointClasses(:Subj11Department :Subj18Department)
DisjointClasses(:Subj11Department :Subj19Department)
DisjointClasses(:Subj11Department :Subj1Department)
DisjointClasses(:Subj11Department :Subj20Department)
DisjointClasses(:Subj11Department :Subj2Department)
DisjointClasses(:Subj11Department :Subj3Department)
DisjointClasses(:Subj11Department :Subj4Department)
DisjointClasses(:Subj11Department :Subj5Department)
DisjointClasses(:Subj11Department :Subj6Department)
DisjointClasses(:Subj11Department :Subj7Department)
DisjointClasses(:Subj11Department :Subj8Department)
DisjointClasses(:Subj11Department :Subj9Department)
DisjointClasses(:Subj11Professor :Subj10Professor)
DisjointClasses(:Subj11Professor :Subj12Professor)
DisjointClasses(:Subj11Professor :Subj13Professor)
DisjointClasses(:Subj11Professor :Subj14Professor)
DisjointClasses(:Subj11Professor :Subj15Professor)
DisjointClasses(:Subj11Professor :Subj16Professor)
DisjointClasses(:Subj11Professor :Subj17Professor)
DisjointClasses(:Subj11Professor :Subj18Professor)
DisjointClasses(:Subj11Professor :Subj19Professor)
DisjointClasses(:Subj11Professor :Subj1Professor)
DisjointClasses(:Subj11Professor :Subj20Professor)
DisjointClasses(:Subj11Professor :Subj2Professor)
DisjointClasses(:Subj11Professor :Subj3Professor)
DisjointClasses(:Subj11Professor :Subj4Professor)
DisjointClasses(:Subj11Professor :Subj5Professor)
DisjointClasses(:Subj11Professor :Subj6Professor)
DisjointClasses(:Subj11Professor :Subj7Professor)
DisjointClasses(:Subj11Professor :Subj8Professor)
DisjointClasses(:Subj11Professor :Subj9Professor)
DisjointClasses(:Subj12Course :Subj10Course)
DisjointClasses(:Subj12Course :Subj11Course)
DisjointClasses(:Subj12Course :Subj13Course)
DisjointClasses(:Subj12Course :Subj14Course)
DisjointClasses(:Subj12Course :Subj15Course)
DisjointClasses(:Subj12Course :Subj16Course)
DisjointClasses(:Subj12Course :Subj17Course)
DisjointClasses(:Subj12Course :Subj18Course)
DisjointClasses(:Subj12Course :Subj19Course)
DisjointClasses(:Subj12Course :Subj1Course)
DisjointClasses(:Subj12Course :Subj20Course)
DisjointClasses(:Subj12Course :Subj2Course)
DisjointClasses(:Subj12Course :Subj3Course)
DisjointClasses(:Subj12Course :Subj4Course)
DisjointClasses(:Subj12Course :Subj5Course)
DisjointClasses(:Subj12Course :Subj6Course)
DisjointClasses(:Subj12Course :Subj7Course)
DisjointClasses(:Subj12Course :Subj8Course)
DisjointClasses(:Subj12Course :Subj9Course)
DisjointClasses(:Subj12Department :Subj10Department)
DisjointClasses(:Subj12Department :Subj11Department)
DisjointClasses(:Subj12Department :Subj13Department)
DisjointClasses(:Subj12Department :Subj14Department)
DisjointClasses(:Subj12Department :Subj15Department)
DisjointClasses(:Subj12Department :Subj16Department)
DisjointClasses(:Subj12Department :Subj17Department)
DisjointClasses(:Subj12Department :Subj18Department)
DisjointClasses(:Subj12Department :Subj19Department)
DisjointClasses(:Subj12Department :Subj1Department)
DisjointClasses(:Subj12Department :Subj20Department)
DisjointClasses(:Subj12Department :Subj2Department)
DisjointClasses(:Subj12Department :Subj3Department)
DisjointClasses(:Subj12Department :Subj4Department)
DisjointClasses(:Subj12Department :Subj5Department)
DisjointClasses(:Subj12Department :Subj6Department)
DisjointClasses(:Subj12Department :Subj7Department)
DisjointClasses(:Subj12Department :Subj8Department)
DisjointClasses(:Subj12Department :Subj9Department)
DisjointClasses(:Subj12Professor :Subj10Professor)
DisjointClasses(:Subj12Professor :Subj11Professor)
DisjointClasses(:Subj12Professor :Subj13Professor)
DisjointClasses(:Subj12Professor :Subj14Professor)
DisjointClasses(:Subj12Professor :Subj15Professor)
DisjointClasses(:Subj12Professor :Subj16Professor)
DisjointClasses(:Subj12Professor :Subj17Professor)
DisjointClasses(:Subj12Professor :Subj18Professor)
DisjointClasses(:Subj12Professor :Subj19Professor)
DisjointClasses(:Subj12Professor :Subj1Professor)
DisjointClasses(:Subj12Professor :Subj20Professor)
DisjointClasses(:Subj12Professor :Subj2Professor)
DisjointClasses(:Subj12Professor :Subj3Professor)
DisjointClasses(:Subj12Professor :Subj4Professor)
DisjointClasses(:Subj12Professor :Subj5Professor)
DisjointClasses(:Subj12Professor :Subj6Professor)
DisjointClasses(:Subj12Professor :Subj7Professor)
DisjointClasses(:Subj12Professor :Subj8Professor)
DisjointClasses(:Subj12Professor :Subj9Professor)
DisjointClasses(:Subj13Course :Subj10Course)
DisjointClasses(:Subj13Course :Subj11Course)
DisjointClasses(:Subj13Course :Subj12Course)
DisjointClasses(:Subj13Course :Subj14Course)
DisjointClasses(:Subj13Course :Subj15Course)
DisjointClasses(:Subj13Course :Subj16Course)
DisjointClasses(:Subj13Course :Subj17Course)
DisjointClasses(:Subj13Course :Subj18Course)
DisjointClasses(:Subj13Course :Subj19Course)
DisjointClasses(:Subj13Course :Subj1Course)
DisjointClasses(:Subj13Course :Subj20Course)
DisjointClasses(:Subj13Course :Subj2Course)
DisjointClasses(:Subj13Course :Subj3Course)
DisjointClasses(:Subj13Course :Subj4Course)
DisjointClasses(:Subj13Course :Subj5Course)
DisjointClasses(:Subj13Course :Subj6Course)
DisjointClasses(:Subj13Course :Subj7Course)
DisjointClasses(:Subj13Course :Subj8Course)
DisjointClasses(:Subj13Course :Subj9Course)
DisjointClasses(:Subj13Department :Subj10Department)
DisjointClasses(:Subj13Department :Subj11Department)
DisjointClasses(:Subj13Department :Subj12Department)
DisjointClasses(:Subj13Department :Subj14Department)
DisjointClasses(:Subj13Department :Subj15Department)
DisjointClasses(:Subj13Department :Subj16Department)
DisjointClasses(:Subj13Department :Subj17Department)
DisjointClasses(:Subj13Department :Subj18Department)
DisjointClasses(:Subj13Department :Subj19Department)
DisjointClasses(:Subj13Department :Subj1Department)
DisjointClasses(:Subj13Department :Subj20Department)
DisjointClasses(:Subj13Department :Subj2Department)
DisjointClasses(:Subj13Department :Subj3Department)
DisjointClasses(:Subj13Department :Subj4Department)
DisjointClasses(:Subj13Department :Subj5Department)
DisjointClasses(:Subj13Department :Subj6Department)
DisjointClasses(:Subj13Department :Subj7Department)
DisjointClasses(:Subj13Department :Subj8Department)
DisjointClasses(:Subj13Department :Subj9Department)
DisjointClasses(:Subj13Professor :Subj10Professor)
DisjointClasses(:Subj13Professor :Subj11Professor)
DisjointClasses(:Subj13Professor :Subj12Professor)
DisjointClasses(:Subj13Professor :Subj14Professor)
DisjointClasses(:Subj13Professor :Subj15Professor)
DisjointClasses(:Subj13Professor :Subj16Professor)
DisjointClasses(:Subj13Professor :Subj17Professor)
DisjointClasses(:Subj13Professor :Subj18Professor)
DisjointClasses(:Subj13Professor :Subj19Professor)
DisjointClasses(:Subj13Professor :Subj1Professor)
DisjointClasses(:Subj13Professor :Subj20Professor)
DisjointClasses(:Subj13Professor :Subj2Professor)
```

DisjointClasses(:Subj13Professor :Subj3Professor)
DisjointClasses(:Subj13Professor :Subj4Professor)
DisjointClasses(:Subj13Professor :Subj5Professor)
DisjointClasses(:Subj13Professor :Subj6Professor)
DisjointClasses(:Subj13Professor :Subj7Professor)
DisjointClasses(:Subj13Professor :Subj8Professor)
DisjointClasses(:Subj13Professor :Subj9Professor)
DisjointClasses(:Subj14Course :Subj10Course)
DisjointClasses(:Subj14Course :Subj11Course)
DisjointClasses(:Subj14Course :Subj12Course)
DisjointClasses(:Subj14Course :Subj13Course)
DisjointClasses(:Subj14Course :Subj15Course)
DisjointClasses(:Subj14Course :Subj16Course)
DisjointClasses(:Subj14Course :Subj17Course)
DisjointClasses(:Subj14Course :Subj18Course)
DisjointClasses(:Subj14Course :Subj19Course)
DisjointClasses(:Subj14Course :Subj1Course)
DisjointClasses(:Subj14Course :Subj20Course)
DisjointClasses(:Subj14Course :Subj2Course)
DisjointClasses(:Subj14Course :Subj3Course)
DisjointClasses(:Subj14Course :Subj4Course)
DisjointClasses(:Subj14Course :Subj5Course)
DisjointClasses(:Subj14Course :Subj6Course)
DisjointClasses(:Subj14Course :Subj7Course)
DisjointClasses(:Subj14Course :Subj8Course)
DisjointClasses(:Subj14Course :Subj9Course)
DisjointClasses(:Subj14Department :Subj10Department)
DisjointClasses(:Subj14Department :Subj11Department)
DisjointClasses(:Subj14Department :Subj12Department)
DisjointClasses(:Subj14Department :Subj13Department)
DisjointClasses(:Subj14Department :Subj15Department)
DisjointClasses(:Subj14Department :Subj16Department)
DisjointClasses(:Subj14Department :Subj17Department)
DisjointClasses(:Subj14Department :Subj18Department)
DisjointClasses(:Subj14Department :Subj19Department)
DisjointClasses(:Subj14Department :Subj1Department)
DisjointClasses(:Subj14Department :Subj20Department)
DisjointClasses(:Subj14Department :Subj2Department)
DisjointClasses(:Subj14Department :Subj3Department)
DisjointClasses(:Subj14Department :Subj4Department)
DisjointClasses(:Subj14Department :Subj5Department)
DisjointClasses(:Subj14Department :Subj6Department)
DisjointClasses(:Subj14Department :Subj7Department)
DisjointClasses(:Subj14Department :Subj8Department)
DisjointClasses(:Subj14Department :Subj9Department)
DisjointClasses(:Subj14Professor :Subj10Professor)
DisjointClasses(:Subj14Professor :Subj11Professor)
DisjointClasses(:Subj14Professor :Subj12Professor)
DisjointClasses(:Subj14Professor :Subj13Professor)
DisjointClasses(:Subj14Professor :Subj15Professor)
DisjointClasses(:Subj14Professor :Subj16Professor)
DisjointClasses(:Subj14Professor :Subj17Professor)
DisjointClasses(:Subj14Professor :Subj18Professor)
DisjointClasses(:Subj14Professor :Subj19Professor)
DisjointClasses(:Subj14Professor :Subj1Professor)
DisjointClasses(:Subj14Professor :Subj20Professor)
DisjointClasses(:Subj14Professor :Subj2Professor)
DisjointClasses(:Subj14Professor :Subj3Professor)
DisjointClasses(:Subj14Professor :Subj4Professor)
DisjointClasses(:Subj14Professor :Subj5Professor)
DisjointClasses(:Subj14Professor :Subj6Professor)
DisjointClasses(:Subj14Professor :Subj7Professor)
DisjointClasses(:Subj14Professor :Subj8Professor)
DisjointClasses(:Subj14Professor :Subj9Professor)
DisjointClasses(:Subj15Course :Subj10Course)
DisjointClasses(:Subj15Course :Subj11Course)
DisjointClasses(:Subj15Course :Subj12Course)
DisjointClasses(:Subj15Course :Subj13Course)
DisjointClasses(:Subj15Course :Subj14Course)
DisjointClasses(:Subj15Course :Subj16Course)
DisjointClasses(:Subj15Course :Subj17Course)
DisjointClasses(:Subj15Course :Subj18Course)
DisjointClasses(:Subj15Course :Subj19Course)
DisjointClasses(:Subj15Course :Subj1Course)
DisjointClasses(:Subj15Course :Subj20Course)
DisjointClasses(:Subj15Course :Subj2Course)
DisjointClasses(:Subj15Course :Subj3Course)
DisjointClasses(:Subj15Course :Subj4Course)
DisjointClasses(:Subj15Course :Subj5Course)
DisjointClasses(:Subj15Course :Subj6Course)
DisjointClasses(:Subj15Course :Subj7Course)
DisjointClasses(:Subj15Course :Subj8Course)
DisjointClasses(:Subj15Course :Subj9Course)
DisjointClasses(:Subj15Department :Subj10Department)
DisjointClasses(:Subj15Department :Subj11Department)
DisjointClasses(:Subj15Department :Subj12Department)
DisjointClasses(:Subj15Department :Subj13Department)
DisjointClasses(:Subj15Department :Subj14Department)
DisjointClasses(:Subj15Department :Subj16Department)
DisjointClasses(:Subj15Department :Subj17Department)
DisjointClasses(:Subj15Department :Subj18Department)
DisjointClasses(:Subj15Department :Subj19Department)
DisjointClasses(:Subj15Department :Subj1Department)
DisjointClasses(:Subj15Department :Subj20Department)
DisjointClasses(:Subj15Department :Subj2Department)
DisjointClasses(:Subj15Department :Subj3Department)
DisjointClasses(:Subj15Department :Subj4Department)
DisjointClasses(:Subj15Department :Subj5Department)
DisjointClasses(:Subj15Department :Subj6Department)
DisjointClasses(:Subj15Department :Subj7Department)
DisjointClasses(:Subj15Department :Subj8Department)
DisjointClasses(:Subj15Department :Subj9Department)
DisjointClasses(:Subj15Professor :Subj10Professor)
DisjointClasses(:Subj15Professor :Subj11Professor)
DisjointClasses(:Subj15Professor :Subj12Professor)
DisjointClasses(:Subj15Professor :Subj13Professor)
DisjointClasses(:Subj15Professor :Subj14Professor)
DisjointClasses(:Subj15Professor :Subj16Professor)
DisjointClasses(:Subj15Professor :Subj17Professor)
DisjointClasses(:Subj15Professor :Subj18Professor)
DisjointClasses(:Subj15Professor :Subj19Professor)
DisjointClasses(:Subj15Professor :Subj1Professor)
DisjointClasses(:Subj15Professor :Subj20Professor)
DisjointClasses(:Subj15Professor :Subj2Professor)
DisjointClasses(:Subj15Professor :Subj3Professor)
DisjointClasses(:Subj15Professor :Subj4Professor)
DisjointClasses(:Subj15Professor :Subj5Professor)
DisjointClasses(:Subj15Professor :Subj6Professor)
DisjointClasses(:Subj15Professor :Subj7Professor)
DisjointClasses(:Subj15Professor :Subj8Professor)
DisjointClasses(:Subj15Professor :Subj9Professor)
DisjointClasses(:Subj16Course :Subj10Course)
DisjointClasses(:Subj16Course :Subj11Course)
DisjointClasses(:Subj16Course :Subj12Course)
DisjointClasses(:Subj16Course :Subj13Course)
DisjointClasses(:Subj16Course :Subj14Course)
DisjointClasses(:Subj16Course :Subj15Course)
DisjointClasses(:Subj16Course :Subj17Course)
DisjointClasses(:Subj16Course :Subj18Course)
DisjointClasses(:Subj16Course :Subj19Course)
DisjointClasses(:Subj16Course :Subj1Course)
DisjointClasses(:Subj16Course :Subj20Course)
DisjointClasses(:Subj16Course :Subj2Course)
DisjointClasses(:Subj16Course :Subj3Course)
DisjointClasses(:Subj16Course :Subj4Course)
DisjointClasses(:Subj16Course :Subj5Course)
DisjointClasses(:Subj16Course :Subj6Course)
DisjointClasses(:Subj16Course :Subj7Course)
DisjointClasses(:Subj16Course :Subj8Course)
DisjointClasses(:Subj16Course :Subj9Course)
DisjointClasses(:Subj16Department :Subj10Department)
DisjointClasses(:Subj16Department :Subj11Department)
DisjointClasses(:Subj16Department :Subj12Department)
DisjointClasses(:Subj16Department :Subj13Department)
DisjointClasses(:Subj16Department :Subj14Department)
DisjointClasses(:Subj16Department :Subj15Department)
DisjointClasses(:Subj16Department :Subj17Department)
DisjointClasses(:Subj16Department :Subj18Department)
DisjointClasses(:Subj16Department :Subj19Department)
DisjointClasses(:Subj16Department :Subj1Department)
DisjointClasses(:Subj16Department :Subj20Department)
DisjointClasses(:Subj16Department :Subj2Department)
DisjointClasses(:Subj16Department :Subj3Department)
DisjointClasses(:Subj16Department :Subj4Department)
DisjointClasses(:Subj16Department :Subj5Department)
DisjointClasses(:Subj16Department :Subj6Department)
DisjointClasses(:Subj16Department :Subj7Department)
DisjointClasses(:Subj16Department :Subj8Department)
DisjointClasses(:Subj16Department :Subj9Department)
DisjointClasses(:Subj16Professor :Subj10Professor)
DisjointClasses(:Subj16Professor :Subj11Professor)
DisjointClasses(:Subj16Professor :Subj12Professor)
DisjointClasses(:Subj16Professor :Subj13Professor)
DisjointClasses(:Subj16Professor :Subj14Professor)
DisjointClasses(:Subj16Professor :Subj15Professor)
DisjointClasses(:Subj16Professor :Subj17Professor)
DisjointClasses(:Subj16Professor :Subj18Professor)
DisjointClasses(:Subj16Professor :Subj19Professor)
DisjointClasses(:Subj16Professor :Subj1Professor)
DisjointClasses(:Subj16Professor :Subj20Professor)
DisjointClasses(:Subj16Professor :Subj2Professor)
DisjointClasses(:Subj16Professor :Subj3Professor)
DisjointClasses(:Subj16Professor :Subj4Professor)
DisjointClasses(:Subj16Professor :Subj5Professor)
DisjointClasses(:Subj16Professor :Subj6Professor)
DisjointClasses(:Subj16Professor :Subj7Professor)
DisjointClasses(:Subj16Professor :Subj8Professor)
DisjointClasses(:Subj16Professor :Subj9Professor)
DisjointClasses(:Subj17Course :Subj10Course)
DisjointClasses(:Subj17Course :Subj11Course)
DisjointClasses(:Subj17Course :Subj12Course)
DisjointClasses(:Subj17Course :Subj13Course)
DisjointClasses(:Subj17Course :Subj14Course)
DisjointClasses(:Subj17Course :Subj15Course)

```
DisjointClasses(:Subj17Course :Subj16Course)
DisjointClasses(:Subj17Course :Subj18Course)
DisjointClasses(:Subj17Course :Subj19Course)
DisjointClasses(:Subj17Course :Subj1Course)
DisjointClasses(:Subj17Course :Subj20Course)
DisjointClasses(:Subj17Course :Subj2Course)
DisjointClasses(:Subj17Course :Subj3Course)
DisjointClasses(:Subj17Course :Subj4Course)
DisjointClasses(:Subj17Course :Subj5Course)
DisjointClasses(:Subj17Course :Subj6Course)
DisjointClasses(:Subj17Course :Subj7Course)
DisjointClasses(:Subj17Course :Subj8Course)
DisjointClasses(:Subj17Course :Subj9Course)
DisjointClasses(:Subj17Department :Subj10Department)
DisjointClasses(:Subj17Department :Subj11Department)
DisjointClasses(:Subj17Department :Subj12Department)
DisjointClasses(:Subj17Department :Subj13Department)
DisjointClasses(:Subj17Department :Subj14Department)
DisjointClasses(:Subj17Department :Subj15Department)
DisjointClasses(:Subj17Department :Subj16Department)
DisjointClasses(:Subj17Department :Subj18Department)
DisjointClasses(:Subj17Department :Subj19Department)
DisjointClasses(:Subj17Department :Subj1Department)
DisjointClasses(:Subj17Department :Subj20Department)
DisjointClasses(:Subj17Department :Subj2Department)
DisjointClasses(:Subj17Department :Subj3Department)
DisjointClasses(:Subj17Department :Subj4Department)
DisjointClasses(:Subj17Department :Subj5Department)
DisjointClasses(:Subj17Department :Subj6Department)
DisjointClasses(:Subj17Department :Subj7Department)
DisjointClasses(:Subj17Department :Subj8Department)
DisjointClasses(:Subj17Department :Subj9Department)
DisjointClasses(:Subj17Professor :Subj10Professor)
DisjointClasses(:Subj17Professor :Subj11Professor)
DisjointClasses(:Subj17Professor :Subj12Professor)
DisjointClasses(:Subj17Professor :Subj13Professor)
DisjointClasses(:Subj17Professor :Subj14Professor)
DisjointClasses(:Subj17Professor :Subj15Professor)
DisjointClasses(:Subj17Professor :Subj16Professor)
DisjointClasses(:Subj17Professor :Subj18Professor)
DisjointClasses(:Subj17Professor :Subj19Professor)
DisjointClasses(:Subj17Professor :Subj1Professor)
DisjointClasses(:Subj17Professor :Subj20Professor)
DisjointClasses(:Subj17Professor :Subj2Professor)
DisjointClasses(:Subj17Professor :Subj3Professor)
DisjointClasses(:Subj17Professor :Subj4Professor)
DisjointClasses(:Subj17Professor :Subj5Professor)
DisjointClasses(:Subj17Professor :Subj6Professor)
DisjointClasses(:Subj17Professor :Subj7Professor)
DisjointClasses(:Subj17Professor :Subj8Professor)
DisjointClasses(:Subj17Professor :Subj9Professor)
DisjointClasses(:Subj18Course :Subj10Course)
DisjointClasses(:Subj18Course :Subj11Course)
DisjointClasses(:Subj18Course :Subj12Course)
DisjointClasses(:Subj18Course :Subj13Course)
DisjointClasses(:Subj18Course :Subj14Course)
DisjointClasses(:Subj18Course :Subj15Course)
DisjointClasses(:Subj18Course :Subj16Course)
DisjointClasses(:Subj18Course :Subj17Course)
DisjointClasses(:Subj18Course :Subj19Course)
DisjointClasses(:Subj18Course :Subj1Course)
DisjointClasses(:Subj18Course :Subj20Course)
DisjointClasses(:Subj18Course :Subj2Course)
DisjointClasses(:Subj18Course :Subj3Course)
DisjointClasses(:Subj18Course :Subj4Course)
DisjointClasses(:Subj18Course :Subj5Course)
DisjointClasses(:Subj18Course :Subj6Course)
DisjointClasses(:Subj18Course :Subj7Course)
DisjointClasses(:Subj18Course :Subj8Course)
DisjointClasses(:Subj18Course :Subj9Course)
DisjointClasses(:Subj18Department :Subj10Department)
DisjointClasses(:Subj18Department :Subj11Department)
DisjointClasses(:Subj18Department :Subj12Department)
DisjointClasses(:Subj18Department :Subj13Department)
DisjointClasses(:Subj18Department :Subj14Department)
DisjointClasses(:Subj18Department :Subj15Department)
DisjointClasses(:Subj18Department :Subj16Department)
DisjointClasses(:Subj18Department :Subj17Department)
DisjointClasses(:Subj18Department :Subj19Department)
DisjointClasses(:Subj18Department :Subj1Department)
DisjointClasses(:Subj18Department :Subj20Department)
DisjointClasses(:Subj18Department :Subj2Department)
DisjointClasses(:Subj18Department :Subj3Department)
DisjointClasses(:Subj18Department :Subj4Department)
DisjointClasses(:Subj18Department :Subj5Department)
DisjointClasses(:Subj18Department :Subj6Department)
DisjointClasses(:Subj18Department :Subj7Department)
DisjointClasses(:Subj18Department :Subj8Department)
DisjointClasses(:Subj18Department :Subj9Department)
DisjointClasses(:Subj18Professor :Subj10Professor)
DisjointClasses(:Subj18Professor :Subj11Professor)
DisjointClasses(:Subj18Professor :Subj12Professor)
DisjointClasses(:Subj18Professor :Subj13Professor)
DisjointClasses(:Subj18Professor :Subj14Professor)
DisjointClasses(:Subj18Professor :Subj15Professor)
DisjointClasses(:Subj18Professor :Subj16Professor)
DisjointClasses(:Subj18Professor :Subj17Professor)
DisjointClasses(:Subj18Professor :Subj19Professor)
DisjointClasses(:Subj18Professor :Subj1Professor)
DisjointClasses(:Subj18Professor :Subj20Professor)
DisjointClasses(:Subj18Professor :Subj2Professor)
DisjointClasses(:Subj18Professor :Subj3Professor)
DisjointClasses(:Subj18Professor :Subj4Professor)
DisjointClasses(:Subj18Professor :Subj5Professor)
DisjointClasses(:Subj18Professor :Subj6Professor)
DisjointClasses(:Subj18Professor :Subj7Professor)
DisjointClasses(:Subj18Professor :Subj8Professor)
DisjointClasses(:Subj18Professor :Subj9Professor)
DisjointClasses(:Subj19Course :Subj10Course)
DisjointClasses(:Subj19Course :Subj11Course)
DisjointClasses(:Subj19Course :Subj12Course)
DisjointClasses(:Subj19Course :Subj13Course)
DisjointClasses(:Subj19Course :Subj14Course)
DisjointClasses(:Subj19Course :Subj15Course)
DisjointClasses(:Subj19Course :Subj16Course)
DisjointClasses(:Subj19Course :Subj17Course)
DisjointClasses(:Subj19Course :Subj18Course)
DisjointClasses(:Subj19Course :Subj1Course)
DisjointClasses(:Subj19Course :Subj20Course)
DisjointClasses(:Subj19Course :Subj2Course)
DisjointClasses(:Subj19Course :Subj3Course)
DisjointClasses(:Subj19Course :Subj4Course)
DisjointClasses(:Subj19Course :Subj5Course)
DisjointClasses(:Subj19Course :Subj6Course)
DisjointClasses(:Subj19Course :Subj7Course)
DisjointClasses(:Subj19Course :Subj8Course)
DisjointClasses(:Subj19Course :Subj9Course)
DisjointClasses(:Subj19Department :Subj10Department)
DisjointClasses(:Subj19Department :Subj11Department)
DisjointClasses(:Subj19Department :Subj12Department)
DisjointClasses(:Subj19Department :Subj13Department)
DisjointClasses(:Subj19Department :Subj14Department)
DisjointClasses(:Subj19Department :Subj15Department)
DisjointClasses(:Subj19Department :Subj16Department)
DisjointClasses(:Subj19Department :Subj17Department)
DisjointClasses(:Subj19Department :Subj18Department)
DisjointClasses(:Subj19Department :Subj1Department)
DisjointClasses(:Subj19Department :Subj20Department)
DisjointClasses(:Subj19Department :Subj2Department)
DisjointClasses(:Subj19Department :Subj3Department)
DisjointClasses(:Subj19Department :Subj4Department)
DisjointClasses(:Subj19Department :Subj5Department)
DisjointClasses(:Subj19Department :Subj6Department)
DisjointClasses(:Subj19Department :Subj7Department)
DisjointClasses(:Subj19Department :Subj8Department)
DisjointClasses(:Subj19Department :Subj9Department)
DisjointClasses(:Subj19Professor :Subj10Professor)
DisjointClasses(:Subj19Professor :Subj11Professor)
DisjointClasses(:Subj19Professor :Subj12Professor)
DisjointClasses(:Subj19Professor :Subj13Professor)
DisjointClasses(:Subj19Professor :Subj14Professor)
DisjointClasses(:Subj19Professor :Subj15Professor)
DisjointClasses(:Subj19Professor :Subj16Professor)
DisjointClasses(:Subj19Professor :Subj17Professor)
DisjointClasses(:Subj19Professor :Subj18Professor)
DisjointClasses(:Subj19Professor :Subj1Professor)
DisjointClasses(:Subj19Professor :Subj20Professor)
DisjointClasses(:Subj19Professor :Subj2Professor)
DisjointClasses(:Subj19Professor :Subj3Professor)
DisjointClasses(:Subj19Professor :Subj4Professor)
DisjointClasses(:Subj19Professor :Subj5Professor)
DisjointClasses(:Subj19Professor :Subj6Professor)
DisjointClasses(:Subj19Professor :Subj7Professor)
DisjointClasses(:Subj19Professor :Subj8Professor)
DisjointClasses(:Subj19Professor :Subj9Professor)
DisjointClasses(:Subj1Course :Subj10Course)
DisjointClasses(:Subj1Course :Subj11Course)
DisjointClasses(:Subj1Course :Subj12Course)
DisjointClasses(:Subj1Course :Subj13Course)
DisjointClasses(:Subj1Course :Subj14Course)
DisjointClasses(:Subj1Course :Subj15Course)
DisjointClasses(:Subj1Course :Subj16Course)
DisjointClasses(:Subj1Course :Subj17Course)
DisjointClasses(:Subj1Course :Subj18Course)
DisjointClasses(:Subj1Course :Subj19Course)
DisjointClasses(:Subj1Course :Subj20Course)
DisjointClasses(:Subj1Course :Subj2Course)
DisjointClasses(:Subj1Course :Subj3Course)
DisjointClasses(:Subj1Course :Subj4Course)
DisjointClasses(:Subj1Course :Subj5Course)
DisjointClasses(:Subj1Course :Subj6Course)
DisjointClasses(:Subj1Course :Subj7Course)
DisjointClasses(:Subj1Course :Subj8Course)
DisjointClasses(:Subj1Course :Subj9Course)
```

```
DisjointClasses(:Subj1Department :Subj10Department)
DisjointClasses(:Subj1Department :Subj11Department)
DisjointClasses(:Subj1Department :Subj12Department)
DisjointClasses(:Subj1Department :Subj13Department)
DisjointClasses(:Subj1Department :Subj14Department)
DisjointClasses(:Subj1Department :Subj15Department)
DisjointClasses(:Subj1Department :Subj16Department)
DisjointClasses(:Subj1Department :Subj17Department)
DisjointClasses(:Subj1Department :Subj18Department)
DisjointClasses(:Subj1Department :Subj19Department)
DisjointClasses(:Subj1Department :Subj20Department)
DisjointClasses(:Subj1Department :Subj2Department)
DisjointClasses(:Subj1Department :Subj3Department)
DisjointClasses(:Subj1Department :Subj4Department)
DisjointClasses(:Subj1Department :Subj5Department)
DisjointClasses(:Subj1Department :Subj6Department)
DisjointClasses(:Subj1Department :Subj7Department)
DisjointClasses(:Subj1Department :Subj8Department)
DisjointClasses(:Subj1Department :Subj9Department)
DisjointClasses(:Subj1Professor :Subj10Professor)
DisjointClasses(:Subj1Professor :Subj11Professor)
DisjointClasses(:Subj1Professor :Subj12Professor)
DisjointClasses(:Subj1Professor :Subj13Professor)
DisjointClasses(:Subj1Professor :Subj14Professor)
DisjointClasses(:Subj1Professor :Subj15Professor)
DisjointClasses(:Subj1Professor :Subj16Professor)
DisjointClasses(:Subj1Professor :Subj17Professor)
DisjointClasses(:Subj1Professor :Subj18Professor)
DisjointClasses(:Subj1Professor :Subj19Professor)
DisjointClasses(:Subj1Professor :Subj20Professor)
DisjointClasses(:Subj1Professor :Subj2Professor)
DisjointClasses(:Subj1Professor :Subj3Professor)
DisjointClasses(:Subj1Professor :Subj4Professor)
DisjointClasses(:Subj1Professor :Subj5Professor)
DisjointClasses(:Subj1Professor :Subj6Professor)
DisjointClasses(:Subj1Professor :Subj7Professor)
DisjointClasses(:Subj1Professor :Subj8Professor)
DisjointClasses(:Subj1Professor :Subj9Professor)
DisjointClasses(:Subj20Course :Subj10Course)
DisjointClasses(:Subj20Course :Subj11Course)
DisjointClasses(:Subj20Course :Subj12Course)
DisjointClasses(:Subj20Course :Subj13Course)
DisjointClasses(:Subj20Course :Subj14Course)
DisjointClasses(:Subj20Course :Subj15Course)
DisjointClasses(:Subj20Course :Subj16Course)
DisjointClasses(:Subj20Course :Subj17Course)
DisjointClasses(:Subj20Course :Subj18Course)
DisjointClasses(:Subj20Course :Subj19Course)
DisjointClasses(:Subj20Course :Subj1Course)
DisjointClasses(:Subj20Course :Subj2Course)
DisjointClasses(:Subj20Course :Subj3Course)
DisjointClasses(:Subj20Course :Subj4Course)
DisjointClasses(:Subj20Course :Subj5Course)
DisjointClasses(:Subj20Course :Subj6Course)
DisjointClasses(:Subj20Course :Subj7Course)
DisjointClasses(:Subj20Course :Subj8Course)
DisjointClasses(:Subj20Course :Subj9Course)
DisjointClasses(:Subj20Department :Subj10Department)
DisjointClasses(:Subj20Department :Subj11Department)
DisjointClasses(:Subj20Department :Subj12Department)
DisjointClasses(:Subj20Department :Subj13Department)
DisjointClasses(:Subj20Department :Subj14Department)
DisjointClasses(:Subj20Department :Subj15Department)
DisjointClasses(:Subj20Department :Subj16Department)
DisjointClasses(:Subj20Department :Subj17Department)
DisjointClasses(:Subj20Department :Subj18Department)
DisjointClasses(:Subj20Department :Subj19Department)
DisjointClasses(:Subj20Department :Subj1Department)
DisjointClasses(:Subj20Department :Subj2Department)
DisjointClasses(:Subj20Department :Subj3Department)
DisjointClasses(:Subj20Department :Subj4Department)
DisjointClasses(:Subj20Department :Subj5Department)
DisjointClasses(:Subj20Department :Subj6Department)
DisjointClasses(:Subj20Department :Subj7Department)
DisjointClasses(:Subj20Department :Subj8Department)
DisjointClasses(:Subj20Department :Subj9Department)
DisjointClasses(:Subj20Professor :Subj10Professor)
DisjointClasses(:Subj20Professor :Subj11Professor)
DisjointClasses(:Subj20Professor :Subj12Professor)
DisjointClasses(:Subj20Professor :Subj13Professor)
DisjointClasses(:Subj20Professor :Subj14Professor)
DisjointClasses(:Subj20Professor :Subj15Professor)
DisjointClasses(:Subj20Professor :Subj16Professor)
DisjointClasses(:Subj20Professor :Subj17Professor)
DisjointClasses(:Subj20Professor :Subj18Professor)
DisjointClasses(:Subj20Professor :Subj19Professor)
DisjointClasses(:Subj20Professor :Subj1Professor)
DisjointClasses(:Subj20Professor :Subj2Professor)
DisjointClasses(:Subj20Professor :Subj3Professor)
DisjointClasses(:Subj20Professor :Subj4Professor)
DisjointClasses(:Subj20Professor :Subj5Professor)
DisjointClasses(:Subj20Professor :Subj6Professor)
DisjointClasses(:Subj20Professor :Subj7Professor)
DisjointClasses(:Subj20Professor :Subj8Professor)
DisjointClasses(:Subj20Professor :Subj9Professor)
DisjointClasses(:Subj2Course :Subj10Course)
DisjointClasses(:Subj2Course :Subj11Course)
DisjointClasses(:Subj2Course :Subj12Course)
DisjointClasses(:Subj2Course :Subj13Course)
DisjointClasses(:Subj2Course :Subj14Course)
DisjointClasses(:Subj2Course :Subj15Course)
DisjointClasses(:Subj2Course :Subj16Course)
DisjointClasses(:Subj2Course :Subj17Course)
DisjointClasses(:Subj2Course :Subj18Course)
DisjointClasses(:Subj2Course :Subj19Course)
DisjointClasses(:Subj2Course :Subj1Course)
DisjointClasses(:Subj2Course :Subj20Course)
DisjointClasses(:Subj2Course :Subj3Course)
DisjointClasses(:Subj2Course :Subj4Course)
DisjointClasses(:Subj2Course :Subj5Course)
DisjointClasses(:Subj2Course :Subj6Course)
DisjointClasses(:Subj2Course :Subj7Course)
DisjointClasses(:Subj2Course :Subj8Course)
DisjointClasses(:Subj2Course :Subj9Course)
DisjointClasses(:Subj2Department :Subj10Department)
DisjointClasses(:Subj2Department :Subj11Department)
DisjointClasses(:Subj2Department :Subj12Department)
DisjointClasses(:Subj2Department :Subj13Department)
DisjointClasses(:Subj2Department :Subj14Department)
DisjointClasses(:Subj2Department :Subj15Department)
DisjointClasses(:Subj2Department :Subj16Department)
DisjointClasses(:Subj2Department :Subj17Department)
DisjointClasses(:Subj2Department :Subj18Department)
DisjointClasses(:Subj2Department :Subj19Department)
DisjointClasses(:Subj2Department :Subj1Department)
DisjointClasses(:Subj2Department :Subj20Department)
DisjointClasses(:Subj2Department :Subj3Department)
DisjointClasses(:Subj2Department :Subj4Department)
DisjointClasses(:Subj2Department :Subj5Department)
DisjointClasses(:Subj2Department :Subj6Department)
DisjointClasses(:Subj2Department :Subj7Department)
DisjointClasses(:Subj2Department :Subj8Department)
DisjointClasses(:Subj2Department :Subj9Department)
DisjointClasses(:Subj2Professor :Subj10Professor)
DisjointClasses(:Subj2Professor :Subj11Professor)
DisjointClasses(:Subj2Professor :Subj12Professor)
DisjointClasses(:Subj2Professor :Subj13Professor)
DisjointClasses(:Subj2Professor :Subj14Professor)
DisjointClasses(:Subj2Professor :Subj15Professor)
DisjointClasses(:Subj2Professor :Subj16Professor)
DisjointClasses(:Subj2Professor :Subj17Professor)
DisjointClasses(:Subj2Professor :Subj18Professor)
DisjointClasses(:Subj2Professor :Subj19Professor)
DisjointClasses(:Subj2Professor :Subj1Professor)
DisjointClasses(:Subj2Professor :Subj20Professor)
DisjointClasses(:Subj2Professor :Subj3Professor)
DisjointClasses(:Subj2Professor :Subj4Professor)
DisjointClasses(:Subj2Professor :Subj5Professor)
DisjointClasses(:Subj2Professor :Subj6Professor)
DisjointClasses(:Subj2Professor :Subj7Professor)
DisjointClasses(:Subj2Professor :Subj8Professor)
DisjointClasses(:Subj2Professor :Subj9Professor)
DisjointClasses(:Subj3Course :Subj10Course)
DisjointClasses(:Subj3Course :Subj11Course)
DisjointClasses(:Subj3Course :Subj12Course)
DisjointClasses(:Subj3Course :Subj13Course)
DisjointClasses(:Subj3Course :Subj14Course)
DisjointClasses(:Subj3Course :Subj15Course)
DisjointClasses(:Subj3Course :Subj16Course)
DisjointClasses(:Subj3Course :Subj17Course)
DisjointClasses(:Subj3Course :Subj18Course)
DisjointClasses(:Subj3Course :Subj19Course)
DisjointClasses(:Subj3Course :Subj1Course)
DisjointClasses(:Subj3Course :Subj20Course)
DisjointClasses(:Subj3Course :Subj2Course)
DisjointClasses(:Subj3Course :Subj4Course)
DisjointClasses(:Subj3Course :Subj5Course)
DisjointClasses(:Subj3Course :Subj6Course)
DisjointClasses(:Subj3Course :Subj7Course)
DisjointClasses(:Subj3Course :Subj8Course)
DisjointClasses(:Subj3Course :Subj9Course)
DisjointClasses(:Subj3Department :Subj10Department)
DisjointClasses(:Subj3Department :Subj11Department)
DisjointClasses(:Subj3Department :Subj12Department)
DisjointClasses(:Subj3Department :Subj13Department)
DisjointClasses(:Subj3Department :Subj14Department)
DisjointClasses(:Subj3Department :Subj15Department)
DisjointClasses(:Subj3Department :Subj16Department)
DisjointClasses(:Subj3Department :Subj17Department)
DisjointClasses(:Subj3Department :Subj18Department)
DisjointClasses(:Subj3Department :Subj19Department)
DisjointClasses(:Subj3Department :Subj1Department)
DisjointClasses(:Subj3Department :Subj20Department)
DisjointClasses(:Subj3Department :Subj2Department)
```

```
DisjointClasses(:Subj3Department :Subj4Department)
DisjointClasses(:Subj3Department :Subj5Department)
DisjointClasses(:Subj3Department :Subj6Department)
DisjointClasses(:Subj3Department :Subj7Department)
DisjointClasses(:Subj3Department :Subj8Department)
DisjointClasses(:Subj3Department :Subj9Department)
DisjointClasses(:Subj3Professor :Subj10Professor)
DisjointClasses(:Subj3Professor :Subj11Professor)
DisjointClasses(:Subj3Professor :Subj12Professor)
DisjointClasses(:Subj3Professor :Subj13Professor)
DisjointClasses(:Subj3Professor :Subj14Professor)
DisjointClasses(:Subj3Professor :Subj15Professor)
DisjointClasses(:Subj3Professor :Subj16Professor)
DisjointClasses(:Subj3Professor :Subj17Professor)
DisjointClasses(:Subj3Professor :Subj18Professor)
DisjointClasses(:Subj3Professor :Subj19Professor)
DisjointClasses(:Subj3Professor :Subj1Professor)
DisjointClasses(:Subj3Professor :Subj20Professor)
DisjointClasses(:Subj3Professor :Subj2Professor)
DisjointClasses(:Subj3Professor :Subj4Professor)
DisjointClasses(:Subj3Professor :Subj5Professor)
DisjointClasses(:Subj3Professor :Subj6Professor)
DisjointClasses(:Subj3Professor :Subj7Professor)
DisjointClasses(:Subj3Professor :Subj8Professor)
DisjointClasses(:Subj3Professor :Subj9Professor)
DisjointClasses(:Subj4Course :Subj10Course)
DisjointClasses(:Subj4Course :Subj11Course)
DisjointClasses(:Subj4Course :Subj12Course)
DisjointClasses(:Subj4Course :Subj13Course)
DisjointClasses(:Subj4Course :Subj14Course)
DisjointClasses(:Subj4Course :Subj15Course)
DisjointClasses(:Subj4Course :Subj16Course)
DisjointClasses(:Subj4Course :Subj17Course)
DisjointClasses(:Subj4Course :Subj18Course)
DisjointClasses(:Subj4Course :Subj19Course)
DisjointClasses(:Subj4Course :Subj1Course)
DisjointClasses(:Subj4Course :Subj20Course)
DisjointClasses(:Subj4Course :Subj2Course)
DisjointClasses(:Subj4Course :Subj3Course)
DisjointClasses(:Subj4Course :Subj5Course)
DisjointClasses(:Subj4Course :Subj6Course)
DisjointClasses(:Subj4Course :Subj7Course)
DisjointClasses(:Subj4Course :Subj8Course)
DisjointClasses(:Subj4Course :Subj9Course)
DisjointClasses(:Subj4Department :Subj10Department)
DisjointClasses(:Subj4Department :Subj11Department)
DisjointClasses(:Subj4Department :Subj12Department)
DisjointClasses(:Subj4Department :Subj13Department)
DisjointClasses(:Subj4Department :Subj14Department)
DisjointClasses(:Subj4Department :Subj15Department)
DisjointClasses(:Subj4Department :Subj16Department)
DisjointClasses(:Subj4Department :Subj17Department)
DisjointClasses(:Subj4Department :Subj18Department)
DisjointClasses(:Subj4Department :Subj19Department)
DisjointClasses(:Subj4Department :Subj1Department)
DisjointClasses(:Subj4Department :Subj20Department)
DisjointClasses(:Subj4Department :Subj2Department)
DisjointClasses(:Subj4Department :Subj3Department)
DisjointClasses(:Subj4Department :Subj5Department)
DisjointClasses(:Subj4Department :Subj6Department)
DisjointClasses(:Subj4Department :Subj7Department)
DisjointClasses(:Subj4Department :Subj8Department)
DisjointClasses(:Subj4Department :Subj9Department)
DisjointClasses(:Subj4Professor :Subj10Professor)
DisjointClasses(:Subj4Professor :Subj11Professor)
DisjointClasses(:Subj4Professor :Subj12Professor)
DisjointClasses(:Subj4Professor :Subj13Professor)
DisjointClasses(:Subj4Professor :Subj14Professor)
DisjointClasses(:Subj4Professor :Subj15Professor)
DisjointClasses(:Subj4Professor :Subj16Professor)
DisjointClasses(:Subj4Professor :Subj17Professor)
DisjointClasses(:Subj4Professor :Subj18Professor)
DisjointClasses(:Subj4Professor :Subj19Professor)
DisjointClasses(:Subj4Professor :Subj1Professor)
DisjointClasses(:Subj4Professor :Subj20Professor)
DisjointClasses(:Subj4Professor :Subj2Professor)
DisjointClasses(:Subj4Professor :Subj3Professor)
DisjointClasses(:Subj4Professor :Subj5Professor)
DisjointClasses(:Subj4Professor :Subj6Professor)
DisjointClasses(:Subj4Professor :Subj7Professor)
DisjointClasses(:Subj4Professor :Subj8Professor)
DisjointClasses(:Subj4Professor :Subj9Professor)
DisjointClasses(:Subj5Course :Subj10Course)
DisjointClasses(:Subj5Course :Subj11Course)
DisjointClasses(:Subj5Course :Subj12Course)
DisjointClasses(:Subj5Course :Subj13Course)
DisjointClasses(:Subj5Course :Subj14Course)
DisjointClasses(:Subj5Course :Subj15Course)
DisjointClasses(:Subj5Course :Subj16Course)
DisjointClasses(:Subj5Course :Subj17Course)
DisjointClasses(:Subj5Course :Subj18Course)
DisjointClasses(:Subj5Course :Subj19Course)
DisjointClasses(:Subj5Course :Subj1Course)
DisjointClasses(:Subj5Course :Subj20Course)
DisjointClasses(:Subj5Course :Subj2Course)
DisjointClasses(:Subj5Course :Subj3Course)
DisjointClasses(:Subj5Course :Subj4Course)
DisjointClasses(:Subj5Course :Subj6Course)
DisjointClasses(:Subj5Course :Subj7Course)
DisjointClasses(:Subj5Course :Subj8Course)
DisjointClasses(:Subj5Course :Subj9Course)
DisjointClasses(:Subj5Department :Subj10Department)
DisjointClasses(:Subj5Department :Subj11Department)
DisjointClasses(:Subj5Department :Subj12Department)
DisjointClasses(:Subj5Department :Subj13Department)
DisjointClasses(:Subj5Department :Subj14Department)
DisjointClasses(:Subj5Department :Subj15Department)
DisjointClasses(:Subj5Department :Subj16Department)
DisjointClasses(:Subj5Department :Subj17Department)
DisjointClasses(:Subj5Department :Subj18Department)
DisjointClasses(:Subj5Department :Subj19Department)
DisjointClasses(:Subj5Department :Subj1Department)
DisjointClasses(:Subj5Department :Subj20Department)
DisjointClasses(:Subj5Department :Subj2Department)
DisjointClasses(:Subj5Department :Subj3Department)
DisjointClasses(:Subj5Department :Subj4Department)
DisjointClasses(:Subj5Department :Subj6Department)
DisjointClasses(:Subj5Department :Subj7Department)
DisjointClasses(:Subj5Department :Subj8Department)
DisjointClasses(:Subj5Department :Subj9Department)
DisjointClasses(:Subj5Professor :Subj10Professor)
DisjointClasses(:Subj5Professor :Subj11Professor)
DisjointClasses(:Subj5Professor :Subj12Professor)
DisjointClasses(:Subj5Professor :Subj13Professor)
DisjointClasses(:Subj5Professor :Subj14Professor)
DisjointClasses(:Subj5Professor :Subj15Professor)
DisjointClasses(:Subj5Professor :Subj16Professor)
DisjointClasses(:Subj5Professor :Subj17Professor)
DisjointClasses(:Subj5Professor :Subj18Professor)
DisjointClasses(:Subj5Professor :Subj19Professor)
DisjointClasses(:Subj5Professor :Subj1Professor)
DisjointClasses(:Subj5Professor :Subj20Professor)
DisjointClasses(:Subj5Professor :Subj2Professor)
DisjointClasses(:Subj5Professor :Subj3Professor)
DisjointClasses(:Subj5Professor :Subj4Professor)
DisjointClasses(:Subj5Professor :Subj6Professor)
DisjointClasses(:Subj5Professor :Subj7Professor)
DisjointClasses(:Subj5Professor :Subj8Professor)
DisjointClasses(:Subj5Professor :Subj9Professor)
DisjointClasses(:Subj6Course :Subj10Course)
DisjointClasses(:Subj6Course :Subj11Course)
DisjointClasses(:Subj6Course :Subj12Course)
DisjointClasses(:Subj6Course :Subj13Course)
DisjointClasses(:Subj6Course :Subj14Course)
DisjointClasses(:Subj6Course :Subj15Course)
DisjointClasses(:Subj6Course :Subj16Course)
DisjointClasses(:Subj6Course :Subj17Course)
DisjointClasses(:Subj6Course :Subj18Course)
DisjointClasses(:Subj6Course :Subj19Course)
DisjointClasses(:Subj6Course :Subj1Course)
DisjointClasses(:Subj6Course :Subj20Course)
DisjointClasses(:Subj6Course :Subj2Course)
DisjointClasses(:Subj6Course :Subj3Course)
DisjointClasses(:Subj6Course :Subj4Course)
DisjointClasses(:Subj6Course :Subj5Course)
DisjointClasses(:Subj6Course :Subj7Course)
DisjointClasses(:Subj6Course :Subj8Course)
DisjointClasses(:Subj6Course :Subj9Course)
DisjointClasses(:Subj6Department :Subj10Department)
DisjointClasses(:Subj6Department :Subj11Department)
DisjointClasses(:Subj6Department :Subj12Department)
DisjointClasses(:Subj6Department :Subj13Department)
DisjointClasses(:Subj6Department :Subj14Department)
DisjointClasses(:Subj6Department :Subj15Department)
DisjointClasses(:Subj6Department :Subj16Department)
DisjointClasses(:Subj6Department :Subj17Department)
DisjointClasses(:Subj6Department :Subj18Department)
DisjointClasses(:Subj6Department :Subj19Department)
DisjointClasses(:Subj6Department :Subj1Department)
DisjointClasses(:Subj6Department :Subj20Department)
DisjointClasses(:Subj6Department :Subj2Department)
DisjointClasses(:Subj6Department :Subj3Department)
DisjointClasses(:Subj6Department :Subj4Department)
DisjointClasses(:Subj6Department :Subj5Department)
DisjointClasses(:Subj6Department :Subj7Department)
DisjointClasses(:Subj6Department :Subj8Department)
DisjointClasses(:Subj6Department :Subj9Department)
DisjointClasses(:Subj6Professor :Subj10Professor)
DisjointClasses(:Subj6Professor :Subj11Professor)
DisjointClasses(:Subj6Professor :Subj12Professor)
DisjointClasses(:Subj6Professor :Subj13Professor)
DisjointClasses(:Subj6Professor :Subj14Professor)
DisjointClasses(:Subj6Professor :Subj15Professor)
DisjointClasses(:Subj6Professor :Subj16Professor)
```

```
DisjointClasses(:Subj6Professor :Subj17Professor)
DisjointClasses(:Subj6Professor :Subj18Professor)
DisjointClasses(:Subj6Professor :Subj19Professor)
DisjointClasses(:Subj6Professor :Subj1Professor)
DisjointClasses(:Subj6Professor :Subj20Professor)
DisjointClasses(:Subj6Professor :Subj2Professor)
DisjointClasses(:Subj6Professor :Subj3Professor)
DisjointClasses(:Subj6Professor :Subj4Professor)
DisjointClasses(:Subj6Professor :Subj5Professor)
DisjointClasses(:Subj6Professor :Subj7Professor)
DisjointClasses(:Subj6Professor :Subj8Professor)
DisjointClasses(:Subj6Professor :Subj9Professor)
DisjointClasses(:Subj7Course :Subj10Course)
DisjointClasses(:Subj7Course :Subj11Course)
DisjointClasses(:Subj7Course :Subj12Course)
DisjointClasses(:Subj7Course :Subj13Course)
DisjointClasses(:Subj7Course :Subj14Course)
DisjointClasses(:Subj7Course :Subj15Course)
DisjointClasses(:Subj7Course :Subj16Course)
DisjointClasses(:Subj7Course :Subj17Course)
DisjointClasses(:Subj7Course :Subj18Course)
DisjointClasses(:Subj7Course :Subj19Course)
DisjointClasses(:Subj7Course :Subj1Course)
DisjointClasses(:Subj7Course :Subj20Course)
DisjointClasses(:Subj7Course :Subj2Course)
DisjointClasses(:Subj7Course :Subj3Course)
DisjointClasses(:Subj7Course :Subj4Course)
DisjointClasses(:Subj7Course :Subj5Course)
DisjointClasses(:Subj7Course :Subj6Course)
DisjointClasses(:Subj7Course :Subj8Course)
DisjointClasses(:Subj7Course :Subj9Course)
DisjointClasses(:Subj7Department :Subj10Department)
DisjointClasses(:Subj7Department :Subj11Department)
DisjointClasses(:Subj7Department :Subj12Department)
DisjointClasses(:Subj7Department :Subj13Department)
DisjointClasses(:Subj7Department :Subj14Department)
DisjointClasses(:Subj7Department :Subj15Department)
DisjointClasses(:Subj7Department :Subj16Department)
DisjointClasses(:Subj7Department :Subj17Department)
DisjointClasses(:Subj7Department :Subj18Department)
DisjointClasses(:Subj7Department :Subj19Department)
DisjointClasses(:Subj7Department :Subj1Department)
DisjointClasses(:Subj7Department :Subj20Department)
DisjointClasses(:Subj7Department :Subj2Department)
DisjointClasses(:Subj7Department :Subj3Department)
DisjointClasses(:Subj7Department :Subj4Department)
DisjointClasses(:Subj7Department :Subj5Department)
DisjointClasses(:Subj7Department :Subj6Department)
DisjointClasses(:Subj7Department :Subj8Department)
DisjointClasses(:Subj7Department :Subj9Department)
DisjointClasses(:Subj7Professor :Subj10Professor)
DisjointClasses(:Subj7Professor :Subj11Professor)
DisjointClasses(:Subj7Professor :Subj12Professor)
DisjointClasses(:Subj7Professor :Subj13Professor)
DisjointClasses(:Subj7Professor :Subj14Professor)
DisjointClasses(:Subj7Professor :Subj15Professor)
DisjointClasses(:Subj7Professor :Subj16Professor)
DisjointClasses(:Subj7Professor :Subj17Professor)
DisjointClasses(:Subj7Professor :Subj18Professor)
DisjointClasses(:Subj7Professor :Subj19Professor)
DisjointClasses(:Subj7Professor :Subj1Professor)
DisjointClasses(:Subj7Professor :Subj20Professor)
DisjointClasses(:Subj7Professor :Subj2Professor)
DisjointClasses(:Subj7Professor :Subj3Professor)
DisjointClasses(:Subj7Professor :Subj4Professor)
DisjointClasses(:Subj7Professor :Subj5Professor)
DisjointClasses(:Subj7Professor :Subj6Professor)
DisjointClasses(:Subj7Professor :Subj8Professor)
DisjointClasses(:Subj7Professor :Subj9Professor)
DisjointClasses(:Subj8Course :Subj10Course)
DisjointClasses(:Subj8Course :Subj11Course)
DisjointClasses(:Subj8Course :Subj12Course)
DisjointClasses(:Subj8Course :Subj13Course)
DisjointClasses(:Subj8Course :Subj14Course)
DisjointClasses(:Subj8Course :Subj15Course)
DisjointClasses(:Subj8Course :Subj16Course)
DisjointClasses(:Subj8Course :Subj17Course)
DisjointClasses(:Subj8Course :Subj18Course)
DisjointClasses(:Subj8Course :Subj19Course)
DisjointClasses(:Subj8Course :Subj1Course)
DisjointClasses(:Subj8Course :Subj20Course)
DisjointClasses(:Subj8Course :Subj2Course)
DisjointClasses(:Subj8Course :Subj3Course)
DisjointClasses(:Subj8Course :Subj4Course)
DisjointClasses(:Subj8Course :Subj5Course)
DisjointClasses(:Subj8Course :Subj6Course)
DisjointClasses(:Subj8Course :Subj7Course)
DisjointClasses(:Subj8Course :Subj9Course)
DisjointClasses(:Subj8Department :Subj10Department)
DisjointClasses(:Subj8Department :Subj11Department)
DisjointClasses(:Subj8Department :Subj12Department)
DisjointClasses(:Subj8Department :Subj13Department)
DisjointClasses(:Subj8Department :Subj14Department)
DisjointClasses(:Subj8Department :Subj15Department)
DisjointClasses(:Subj8Department :Subj16Department)
DisjointClasses(:Subj8Department :Subj17Department)
DisjointClasses(:Subj8Department :Subj18Department)
DisjointClasses(:Subj8Department :Subj19Department)
DisjointClasses(:Subj8Department :Subj1Department)
DisjointClasses(:Subj8Department :Subj20Department)
DisjointClasses(:Subj8Department :Subj2Department)
DisjointClasses(:Subj8Department :Subj3Department)
DisjointClasses(:Subj8Department :Subj4Department)
DisjointClasses(:Subj8Department :Subj5Department)
DisjointClasses(:Subj8Department :Subj6Department)
DisjointClasses(:Subj8Department :Subj7Department)
DisjointClasses(:Subj8Department :Subj9Department)
DisjointClasses(:Subj8Professor :Subj10Professor)
DisjointClasses(:Subj8Professor :Subj11Professor)
DisjointClasses(:Subj8Professor :Subj12Professor)
DisjointClasses(:Subj8Professor :Subj13Professor)
DisjointClasses(:Subj8Professor :Subj14Professor)
DisjointClasses(:Subj8Professor :Subj15Professor)
DisjointClasses(:Subj8Professor :Subj16Professor)
DisjointClasses(:Subj8Professor :Subj17Professor)
DisjointClasses(:Subj8Professor :Subj18Professor)
DisjointClasses(:Subj8Professor :Subj19Professor)
DisjointClasses(:Subj8Professor :Subj1Professor)
DisjointClasses(:Subj8Professor :Subj20Professor)
DisjointClasses(:Subj8Professor :Subj2Professor)
DisjointClasses(:Subj8Professor :Subj3Professor)
DisjointClasses(:Subj8Professor :Subj4Professor)
DisjointClasses(:Subj8Professor :Subj5Professor)
DisjointClasses(:Subj8Professor :Subj6Professor)
DisjointClasses(:Subj8Professor :Subj7Professor)
DisjointClasses(:Subj8Professor :Subj9Professor)
DisjointClasses(:Subj9Course :Subj10Course)
DisjointClasses(:Subj9Course :Subj11Course)
DisjointClasses(:Subj9Course :Subj12Course)
DisjointClasses(:Subj9Course :Subj13Course)
DisjointClasses(:Subj9Course :Subj14Course)
DisjointClasses(:Subj9Course :Subj15Course)
DisjointClasses(:Subj9Course :Subj16Course)
DisjointClasses(:Subj9Course :Subj17Course)
DisjointClasses(:Subj9Course :Subj18Course)
DisjointClasses(:Subj9Course :Subj19Course)
DisjointClasses(:Subj9Course :Subj1Course)
DisjointClasses(:Subj9Course :Subj20Course)
DisjointClasses(:Subj9Course :Subj2Course)
DisjointClasses(:Subj9Course :Subj3Course)
DisjointClasses(:Subj9Course :Subj4Course)
DisjointClasses(:Subj9Course :Subj5Course)
DisjointClasses(:Subj9Course :Subj6Course)
DisjointClasses(:Subj9Course :Subj7Course)
DisjointClasses(:Subj9Course :Subj8Course)
DisjointClasses(:Subj9Department :Subj10Department)
DisjointClasses(:Subj9Department :Subj11Department)
DisjointClasses(:Subj9Department :Subj12Department)
DisjointClasses(:Subj9Department :Subj13Department)
DisjointClasses(:Subj9Department :Subj14Department)
DisjointClasses(:Subj9Department :Subj15Department)
DisjointClasses(:Subj9Department :Subj16Department)
DisjointClasses(:Subj9Department :Subj17Department)
DisjointClasses(:Subj9Department :Subj18Department)
DisjointClasses(:Subj9Department :Subj19Department)
DisjointClasses(:Subj9Department :Subj1Department)
DisjointClasses(:Subj9Department :Subj20Department)
DisjointClasses(:Subj9Department :Subj2Department)
DisjointClasses(:Subj9Department :Subj3Department)
DisjointClasses(:Subj9Department :Subj4Department)
DisjointClasses(:Subj9Department :Subj5Department)
DisjointClasses(:Subj9Department :Subj6Department)
DisjointClasses(:Subj9Department :Subj7Department)
DisjointClasses(:Subj9Department :Subj8Department)
DisjointClasses(:Subj9Professor :Subj10Professor)
DisjointClasses(:Subj9Professor :Subj11Professor)
DisjointClasses(:Subj9Professor :Subj12Professor)
DisjointClasses(:Subj9Professor :Subj13Professor)
DisjointClasses(:Subj9Professor :Subj14Professor)
DisjointClasses(:Subj9Professor :Subj15Professor)
DisjointClasses(:Subj9Professor :Subj16Professor)
DisjointClasses(:Subj9Professor :Subj17Professor)
DisjointClasses(:Subj9Professor :Subj18Professor)
DisjointClasses(:Subj9Professor :Subj19Professor)
DisjointClasses(:Subj9Professor :Subj1Professor)
DisjointClasses(:Subj9Professor :Subj20Professor)
DisjointClasses(:Subj9Professor :Subj2Professor)
DisjointClasses(:Subj9Professor :Subj3Professor)
DisjointClasses(:Subj9Professor :Subj4Professor)
DisjointClasses(:Subj9Professor :Subj5Professor)
DisjointClasses(:Subj9Professor :Subj6Professor)
DisjointClasses(:Subj9Professor :Subj7Professor)
DisjointClasses(:Subj9Professor :Subj8Professor)
DisjointClasses(:SystemsStaff :ClericalStaff)
```

```
DisjointClasses(:TeachingAssistant :Director)
DisjointClasses(:TeachingAssistant :Employee)
DisjointClasses(:TeachingAssistant :chair)
DisjointClasses(:TechnicalReport :ConferencePaper)
DisjointClasses(:TechnicalReport :JournalArticle)
DisjointClasses(:University :College)
DisjointClasses(:University :Department)
DisjointClasses(:University :Institute)
DisjointClasses(:University :Program)
DisjointClasses(:University :ResearchGroup)
DisjointClasses(:UnofficialPublication :Article)
DisjointClasses(:UnofficialPublication :Book)
DisjointClasses(:UnofficialPublication :Manual)
DisjointClasses(:UnofficialPublication :Software)
DisjointClasses(:UnofficialPublication :Specification)
DisjointClasses(:VisitingProfessor :AssistantProfessor)
DisjointClasses(:VisitingProfessor :AssociateProfessor)
DisjointClasses(:VisitingProfessor :Dean)
DisjointClasses(:VisitingProfessor :ExDean)
DisjointClasses(:VisitingProfessor :FullProfessor)
DisjointClasses(:Work :Organization)
DisjointClasses(:Work :Person)
DisjointClasses(:Work :Publication)
DisjointClasses(:Work :Schedule)
DisjointClasses(:chair :GraduateStudent)
DisjointClasses(:chair :ResearchAssistant)
DisjointClasses(:chair :Student)
DisjointClasses(:chair :TeachingAssistant)
DisjointObjectProperties(:advisor :affiliatedOrganizationOf)
DisjointObjectProperties(:advisor :degreeFrom)
DisjointObjectProperties(:advisor :hasAlumnus)
DisjointObjectProperties(:advisor :hasExamRecord)
DisjointObjectProperties(:advisor :hasFaculty)
DisjointObjectProperties(:advisor :isPartOfUniversity)
DisjointObjectProperties(:advisor :listedCourse)
DisjointObjectProperties(:advisor :member)
DisjointObjectProperties(:advisor :memberOf)
DisjointObjectProperties(:advisor :orgPublication)
DisjointObjectProperties(:advisor :publicationAuthor)
DisjointObjectProperties(:advisor :publicationDate)
DisjointObjectProperties(:advisor :publicationResearch)
DisjointObjectProperties(:advisor :researchProject)
DisjointObjectProperties(:advisor :softwareDocumentation)
DisjointObjectProperties(:advisor :softwareVersion)
DisjointObjectProperties(:advisor :subOrganizationOf)
DisjointObjectProperties(:advisor :takesCourse)
DisjointObjectProperties(:advisor :teacherOf)
DisjointObjectProperties(:advisor :teachingAssistantOf)
DisjointObjectProperties(:advisor :tenured)
DisjointObjectProperties(:affiliatedOrganizationOf :advisor)
DisjointObjectProperties(:affiliatedOrganizationOf :degreeFrom)
DisjointObjectProperties(:affiliatedOrganizationOf :hasAlumnus)
DisjointObjectProperties(:affiliatedOrganizationOf :hasExamRecord)
DisjointObjectProperties(:affiliatedOrganizationOf :hasFaculty)
DisjointObjectProperties(:affiliatedOrganizationOf :isPartOfUniversity)
DisjointObjectProperties(:affiliatedOrganizationOf :listedCourse)
DisjointObjectProperties(:affiliatedOrganizationOf :member)
DisjointObjectProperties(:affiliatedOrganizationOf :memberOf)
DisjointObjectProperties(:affiliatedOrganizationOf :orgPublication)
DisjointObjectProperties(:affiliatedOrganizationOf :publicationAuthor)
DisjointObjectProperties(:affiliatedOrganizationOf :publicationDate)
DisjointObjectProperties(:affiliatedOrganizationOf :researchProject)
DisjointObjectProperties(:affiliatedOrganizationOf :softwareDocumentation)
DisjointObjectProperties(:affiliatedOrganizationOf :softwareVersion)
DisjointObjectProperties(:affiliatedOrganizationOf :subOrganizationOf)
DisjointObjectProperties(:affiliatedOrganizationOf :takesCourse)
DisjointObjectProperties(:affiliatedOrganizationOf :teacherOf)
DisjointObjectProperties(:affiliatedOrganizationOf :teachingAssistantOf)
DisjointObjectProperties(:affiliatedOrganizationOf :tenured)
DisjointObjectProperties(:degreeFrom :advisor)
DisjointObjectProperties(:degreeFrom :affiliatedOrganizationOf)
DisjointObjectProperties(:degreeFrom :hasAlumnus)
DisjointObjectProperties(:degreeFrom :hasExamRecord)
DisjointObjectProperties(:degreeFrom :hasFaculty)
DisjointObjectProperties(:degreeFrom :isPartOfUniversity)
DisjointObjectProperties(:degreeFrom :listedCourse)
DisjointObjectProperties(:degreeFrom :member)
DisjointObjectProperties(:degreeFrom :memberOf)
DisjointObjectProperties(:degreeFrom :orgPublication)
DisjointObjectProperties(:degreeFrom :publicationAuthor)
DisjointObjectProperties(:degreeFrom :publicationDate)
DisjointObjectProperties(:degreeFrom :publicationResearch)
DisjointObjectProperties(:degreeFrom :researchProject)
DisjointObjectProperties(:degreeFrom :softwareDocumentation)
DisjointObjectProperties(:degreeFrom :softwareVersion)
DisjointObjectProperties(:degreeFrom :subOrganizationOf)
DisjointObjectProperties(:degreeFrom :takesCourse)
DisjointObjectProperties(:degreeFrom :teacherOf)
DisjointObjectProperties(:degreeFrom :teachingAssistantOf)
DisjointObjectProperties(:degreeFrom :tenured)
DisjointObjectProperties(:hasAlumnus :advisor)
DisjointObjectProperties(:hasAlumnus :affiliatedOrganizationOf)
DisjointObjectProperties(:hasAlumnus :degreeFrom)
DisjointObjectProperties(:hasAlumnus :hasExamRecord)
DisjointObjectProperties(:hasAlumnus :hasFaculty)
DisjointObjectProperties(:hasAlumnus :isPartOfUniversity)
DisjointObjectProperties(:hasAlumnus :listedCourse)
DisjointObjectProperties(:hasAlumnus :member)
DisjointObjectProperties(:hasAlumnus :memberOf)
DisjointObjectProperties(:hasAlumnus :orgPublication)
DisjointObjectProperties(:hasAlumnus :publicationAuthor)
DisjointObjectProperties(:hasAlumnus :publicationDate)
DisjointObjectProperties(:hasAlumnus :publicationResearch)
DisjointObjectProperties(:hasAlumnus :researchProject)
DisjointObjectProperties(:hasAlumnus :softwareDocumentation)
DisjointObjectProperties(:hasAlumnus :softwareVersion)
DisjointObjectProperties(:hasAlumnus :subOrganizationOf)
DisjointObjectProperties(:hasAlumnus :takesCourse)
DisjointObjectProperties(:hasAlumnus :teacherOf)
DisjointObjectProperties(:hasAlumnus :teachingAssistantOf)
DisjointObjectProperties(:hasAlumnus :tenured)
DisjointObjectProperties(:hasExamRecord :advisor)
DisjointObjectProperties(:hasExamRecord :affiliatedOrganizationOf)
DisjointObjectProperties(:hasExamRecord :degreeFrom)
DisjointObjectProperties(:hasExamRecord :hasAlumnus)
DisjointObjectProperties(:hasExamRecord :hasFaculty)
DisjointObjectProperties(:hasExamRecord :isPartOfUniversity)
DisjointObjectProperties(:hasExamRecord :listedCourse)
DisjointObjectProperties(:hasExamRecord :member)
DisjointObjectProperties(:hasExamRecord :memberOf)
DisjointObjectProperties(:hasExamRecord :publicationDate)
DisjointObjectProperties(:hasExamRecord :publicationResearch)
DisjointObjectProperties(:hasExamRecord :researchProject)
DisjointObjectProperties(:hasExamRecord :softwareDocumentation)
DisjointObjectProperties(:hasExamRecord :softwareVersion)
DisjointObjectProperties(:hasExamRecord :subOrganizationOf)
DisjointObjectProperties(:hasExamRecord :takesCourse)
DisjointObjectProperties(:hasExamRecord :teachingAssistantOf)
DisjointObjectProperties(:hasExamRecord :tenured)
DisjointObjectProperties(:hasFaculty :advisor)
DisjointObjectProperties(:hasFaculty :affiliatedOrganizationOf)
DisjointObjectProperties(:hasFaculty :degreeFrom)
DisjointObjectProperties(:hasFaculty :hasAlumnus)
DisjointObjectProperties(:hasFaculty :hasExamRecord)
DisjointObjectProperties(:hasFaculty :isPartOfUniversity)
DisjointObjectProperties(:hasFaculty :listedCourse)
DisjointObjectProperties(:hasFaculty :member)
DisjointObjectProperties(:hasFaculty :memberOf)
DisjointObjectProperties(:hasFaculty :orgPublication)
DisjointObjectProperties(:hasFaculty :publicationAuthor)
DisjointObjectProperties(:hasFaculty :publicationDate)
DisjointObjectProperties(:hasFaculty :publicationResearch)
DisjointObjectProperties(:hasFaculty :researchProject)
DisjointObjectProperties(:hasFaculty :softwareDocumentation)
DisjointObjectProperties(:hasFaculty :softwareVersion)
DisjointObjectProperties(:hasFaculty :subOrganizationOf)
DisjointObjectProperties(:hasFaculty :takesCourse)
DisjointObjectProperties(:hasFaculty :teacherOf)
DisjointObjectProperties(:hasFaculty :teachingAssistantOf)
DisjointObjectProperties(:hasFaculty :tenured)
DisjointObjectProperties(:isPartOfUniversity :advisor)
DisjointObjectProperties(:isPartOfUniversity :affiliatedOrganizationOf)
DisjointObjectProperties(:isPartOfUniversity :degreeFrom)
DisjointObjectProperties(:isPartOfUniversity :hasAlumnus)
DisjointObjectProperties(:isPartOfUniversity :hasExamRecord)
DisjointObjectProperties(:isPartOfUniversity :hasFaculty)
DisjointObjectProperties(:isPartOfUniversity :listedCourse)
DisjointObjectProperties(:isPartOfUniversity :member)
DisjointObjectProperties(:isPartOfUniversity :memberOf)
DisjointObjectProperties(:isPartOfUniversity :orgPublication)
DisjointObjectProperties(:isPartOfUniversity :publicationAuthor)
DisjointObjectProperties(:isPartOfUniversity :publicationDate)
DisjointObjectProperties(:isPartOfUniversity :publicationResearch)
DisjointObjectProperties(:isPartOfUniversity :researchProject)
DisjointObjectProperties(:isPartOfUniversity :softwareDocumentation)
DisjointObjectProperties(:isPartOfUniversity :softwareVersion)
DisjointObjectProperties(:isPartOfUniversity :subOrganizationOf)
DisjointObjectProperties(:isPartOfUniversity :takesCourse)
DisjointObjectProperties(:isPartOfUniversity :teacherOf)
DisjointObjectProperties(:isPartOfUniversity :teachingAssistantOf)
DisjointObjectProperties(:isPartOfUniversity :tenured)
DisjointObjectProperties(:listedCourse :advisor)
DisjointObjectProperties(:listedCourse :affiliatedOrganizationOf)
DisjointObjectProperties(:listedCourse :degreeFrom)
DisjointObjectProperties(:listedCourse :hasAlumnus)
DisjointObjectProperties(:listedCourse :hasExamRecord)
DisjointObjectProperties(:listedCourse :hasFaculty)
DisjointObjectProperties(:listedCourse :isPartOfUniversity)
DisjointObjectProperties(:listedCourse :member)
DisjointObjectProperties(:listedCourse :memberOf)
DisjointObjectProperties(:listedCourse :orgPublication)
DisjointObjectProperties(:listedCourse :publicationAuthor)
DisjointObjectProperties(:listedCourse :publicationDate)
DisjointObjectProperties(:listedCourse :publicationResearch)
```

```
DisjointObjectProperties(:listedCourse :researchProject)
DisjointObjectProperties(:listedCourse :softwareDocumentation)
DisjointObjectProperties(:listedCourse :softwareVersion)
DisjointObjectProperties(:listedCourse :subOrganizationOf)
DisjointObjectProperties(:listedCourse :takesCourse)
DisjointObjectProperties(:listedCourse :teacherOf)
DisjointObjectProperties(:listedCourse :teachingAssistantOf)
DisjointObjectProperties(:listedCourse :tenured)
DisjointObjectProperties(:member :advisor)
DisjointObjectProperties(:member :affiliatedOrganizationOf)
DisjointObjectProperties(:member :degreeFrom)
DisjointObjectProperties(:member :hasAlumnus)
DisjointObjectProperties(:member :hasExamRecord)
DisjointObjectProperties(:member :hasFaculty)
DisjointObjectProperties(:member :isPartOfUniversity)
DisjointObjectProperties(:member :listedCourse)
DisjointObjectProperties(:member :memberOf)
DisjointObjectProperties(:member :orgPublication)
DisjointObjectProperties(:member :publicationAuthor)
DisjointObjectProperties(:member :publicationDate)
DisjointObjectProperties(:member :publicationResearch)
DisjointObjectProperties(:member :researchProject)
DisjointObjectProperties(:member :softwareDocumentation)
DisjointObjectProperties(:member :softwareVersion)
DisjointObjectProperties(:member :subOrganizationOf)
DisjointObjectProperties(:member :takesCourse)
DisjointObjectProperties(:member :teacherOf)
DisjointObjectProperties(:member :teachingAssistantOf)
DisjointObjectProperties(:member :tenured)
DisjointObjectProperties(:memberOf :advisor)
DisjointObjectProperties(:memberOf :affiliatedOrganizationOf)
DisjointObjectProperties(:memberOf :degreeFrom)
DisjointObjectProperties(:memberOf :hasAlumnus)
DisjointObjectProperties(:memberOf :hasExamRecord)
DisjointObjectProperties(:memberOf :hasFaculty)
DisjointObjectProperties(:memberOf :isPartOfUniversity)
DisjointObjectProperties(:memberOf :listedCourse)
DisjointObjectProperties(:memberOf :member)
DisjointObjectProperties(:memberOf :orgPublication)
DisjointObjectProperties(:memberOf :publicationAuthor)
DisjointObjectProperties(:memberOf :publicationDate)
DisjointObjectProperties(:memberOf :publicationResearch)
DisjointObjectProperties(:memberOf :researchProject)
DisjointObjectProperties(:memberOf :softwareDocumentation)
DisjointObjectProperties(:memberOf :softwareVersion)
DisjointObjectProperties(:memberOf :subOrganizationOf)
DisjointObjectProperties(:memberOf :takesCourse)
DisjointObjectProperties(:memberOf :teacherOf)
DisjointObjectProperties(:memberOf :teachingAssistantOf)
DisjointObjectProperties(:memberOf :tenured)
DisjointObjectProperties(:orgPublication :advisor)
DisjointObjectProperties(:orgPublication :affiliatedOrganizationOf)
DisjointObjectProperties(:orgPublication :degreeFrom)
DisjointObjectProperties(:orgPublication :hasAlumnus)
DisjointObjectProperties(:orgPublication :hasFaculty)
DisjointObjectProperties(:orgPublication :isPartOfUniversity)
DisjointObjectProperties(:orgPublication :listedCourse)
DisjointObjectProperties(:orgPublication :member)
DisjointObjectProperties(:orgPublication :memberOf)
DisjointObjectProperties(:orgPublication :publicationAuthor)
DisjointObjectProperties(:orgPublication :publicationDate)
DisjointObjectProperties(:orgPublication :publicationResearch)
DisjointObjectProperties(:orgPublication :researchProject)
DisjointObjectProperties(:orgPublication :softwareDocumentation)
DisjointObjectProperties(:orgPublication :softwareVersion)
DisjointObjectProperties(:orgPublication :subOrganizationOf)
DisjointObjectProperties(:orgPublication :takesCourse)
DisjointObjectProperties(:orgPublication :teacherOf)
DisjointObjectProperties(:orgPublication :teachingAssistantOf)
DisjointObjectProperties(:orgPublication :tenured)
DisjointObjectProperties(:publicationAuthor :advisor)
DisjointObjectProperties(:publicationAuthor :affiliatedOrganizationOf)
DisjointObjectProperties(:publicationAuthor :degreeFrom)
DisjointObjectProperties(:publicationAuthor :hasAlumnus)
DisjointObjectProperties(:publicationAuthor :hasFaculty)
DisjointObjectProperties(:publicationAuthor :isPartOfUniversity)
DisjointObjectProperties(:publicationAuthor :listedCourse)
DisjointObjectProperties(:publicationAuthor :member)
DisjointObjectProperties(:publicationAuthor :memberOf)
DisjointObjectProperties(:publicationAuthor :orgPublication)
DisjointObjectProperties(:publicationAuthor :publicationDate)
DisjointObjectProperties(:publicationAuthor :publicationResearch)
DisjointObjectProperties(:publicationAuthor :researchProject)
DisjointObjectProperties(:publicationAuthor :softwareDocumentation)
DisjointObjectProperties(:publicationAuthor :softwareVersion)
DisjointObjectProperties(:publicationAuthor :subOrganizationOf)
DisjointObjectProperties(:publicationAuthor :takesCourse)
DisjointObjectProperties(:publicationAuthor :teacherOf)
DisjointObjectProperties(:publicationAuthor :teachingAssistantOf)
DisjointObjectProperties(:publicationAuthor :tenured)
DisjointObjectProperties(:publicationDate :advisor)
DisjointObjectProperties(:publicationDate :affiliatedOrganizationOf)
DisjointObjectProperties(:publicationDate :degreeFrom)
DisjointObjectProperties(:publicationDate :hasAlumnus)
DisjointObjectProperties(:publicationDate :hasExamRecord)
DisjointObjectProperties(:publicationDate :hasFaculty)
DisjointObjectProperties(:publicationDate :isPartOfUniversity)
DisjointObjectProperties(:publicationDate :listedCourse)
DisjointObjectProperties(:publicationDate :member)
DisjointObjectProperties(:publicationDate :memberOf)
DisjointObjectProperties(:publicationDate :orgPublication)
DisjointObjectProperties(:publicationDate :publicationAuthor)
DisjointObjectProperties(:publicationDate :publicationResearch)
DisjointObjectProperties(:publicationDate :researchProject)
DisjointObjectProperties(:publicationDate :softwareDocumentation)
DisjointObjectProperties(:publicationDate :softwareVersion)
DisjointObjectProperties(:publicationDate :subOrganizationOf)
DisjointObjectProperties(:publicationDate :takesCourse)
DisjointObjectProperties(:publicationDate :teacherOf)
DisjointObjectProperties(:publicationDate :teachingAssistantOf)
DisjointObjectProperties(:publicationDate :tenured)
DisjointObjectProperties(:publicationResearch :advisor)
DisjointObjectProperties(:publicationResearch :degreeFrom)
DisjointObjectProperties(:publicationResearch :hasAlumnus)
DisjointObjectProperties(:publicationResearch :hasExamRecord)
DisjointObjectProperties(:publicationResearch :hasFaculty)
DisjointObjectProperties(:publicationResearch :isPartOfUniversity)
DisjointObjectProperties(:publicationResearch :listedCourse)
DisjointObjectProperties(:publicationResearch :member)
DisjointObjectProperties(:publicationResearch :memberOf)
DisjointObjectProperties(:publicationResearch :orgPublication)
DisjointObjectProperties(:publicationResearch :publicationAuthor)
DisjointObjectProperties(:publicationResearch :publicationDate)
DisjointObjectProperties(:publicationResearch :researchProject)
DisjointObjectProperties(:publicationResearch :softwareDocumentation)
DisjointObjectProperties(:publicationResearch :softwareVersion)
DisjointObjectProperties(:publicationResearch :subOrganizationOf)
DisjointObjectProperties(:publicationResearch :takesCourse)
DisjointObjectProperties(:publicationResearch :teacherOf)
DisjointObjectProperties(:publicationResearch :teachingAssistantOf)
DisjointObjectProperties(:publicationResearch :tenured)
DisjointObjectProperties(:researchProject :advisor)
DisjointObjectProperties(:researchProject :affiliatedOrganizationOf)
DisjointObjectProperties(:researchProject :degreeFrom)
DisjointObjectProperties(:researchProject :hasAlumnus)
DisjointObjectProperties(:researchProject :hasExamRecord)
DisjointObjectProperties(:researchProject :hasFaculty)
DisjointObjectProperties(:researchProject :isPartOfUniversity)
DisjointObjectProperties(:researchProject :listedCourse)
DisjointObjectProperties(:researchProject :member)
DisjointObjectProperties(:researchProject :memberOf)
DisjointObjectProperties(:researchProject :orgPublication)
DisjointObjectProperties(:researchProject :publicationAuthor)
DisjointObjectProperties(:researchProject :publicationDate)
DisjointObjectProperties(:researchProject :publicationResearch)
DisjointObjectProperties(:researchProject :softwareDocumentation)
DisjointObjectProperties(:researchProject :softwareVersion)
DisjointObjectProperties(:researchProject :subOrganizationOf)
DisjointObjectProperties(:researchProject :takesCourse)
DisjointObjectProperties(:researchProject :teacherOf)
DisjointObjectProperties(:researchProject :teachingAssistantOf)
DisjointObjectProperties(:researchProject :tenured)
DisjointObjectProperties(:softwareDocumentation :advisor)
DisjointObjectProperties(:softwareDocumentation :affiliatedOrganizationOf)
DisjointObjectProperties(:softwareDocumentation :degreeFrom)
DisjointObjectProperties(:softwareDocumentation :hasAlumnus)
DisjointObjectProperties(:softwareDocumentation :hasExamRecord)
DisjointObjectProperties(:softwareDocumentation :hasFaculty)
DisjointObjectProperties(:softwareDocumentation :isPartOfUniversity)
DisjointObjectProperties(:softwareDocumentation :listedCourse)
DisjointObjectProperties(:softwareDocumentation :member)
DisjointObjectProperties(:softwareDocumentation :memberOf)
DisjointObjectProperties(:softwareDocumentation :orgPublication)
DisjointObjectProperties(:softwareDocumentation :publicationAuthor)
DisjointObjectProperties(:softwareDocumentation :publicationDate)
DisjointObjectProperties(:softwareDocumentation :publicationResearch)
DisjointObjectProperties(:softwareDocumentation :researchProject)
DisjointObjectProperties(:softwareDocumentation :softwareVersion)
DisjointObjectProperties(:softwareDocumentation :subOrganizationOf)
DisjointObjectProperties(:softwareDocumentation :takesCourse)
DisjointObjectProperties(:softwareDocumentation :teacherOf)
DisjointObjectProperties(:softwareDocumentation :teachingAssistantOf)
DisjointObjectProperties(:softwareDocumentation :tenured)
DisjointObjectProperties(:softwareVersion :advisor)
DisjointObjectProperties(:softwareVersion :affiliatedOrganizationOf)
DisjointObjectProperties(:softwareVersion :degreeFrom)
DisjointObjectProperties(:softwareVersion :hasAlumnus)
DisjointObjectProperties(:softwareVersion :hasExamRecord)
DisjointObjectProperties(:softwareVersion :hasFaculty)
DisjointObjectProperties(:softwareVersion :isPartOfUniversity)
DisjointObjectProperties(:softwareVersion :listedCourse)
DisjointObjectProperties(:softwareVersion :member)
DisjointObjectProperties(:softwareVersion :memberOf)
DisjointObjectProperties(:softwareVersion :orgPublication)
```

```
DisjointObjectProperties(:softwareVersion :publicationAuthor)
DisjointObjectProperties(:softwareVersion :publicationDate)
DisjointObjectProperties(:softwareVersion :publicationResearch)
DisjointObjectProperties(:softwareVersion :researchProject)
DisjointObjectProperties(:softwareVersion :softwareDocumentation)
DisjointObjectProperties(:softwareVersion :subOrganizationOf)
DisjointObjectProperties(:softwareVersion :takesCourse)
DisjointObjectProperties(:softwareVersion :teacherOf)
DisjointObjectProperties(:softwareVersion :teachingAssistantOf)
DisjointObjectProperties(:softwareVersion :tenured)
DisjointObjectProperties(:subOrganizationOf :advisor)
DisjointObjectProperties(:subOrganizationOf :affiliatedOrganizationOf)
DisjointObjectProperties(:subOrganizationOf :degreeFrom)
DisjointObjectProperties(:subOrganizationOf :hasAlumnus)
DisjointObjectProperties(:subOrganizationOf :hasExamRecord)
DisjointObjectProperties(:subOrganizationOf :hasFaculty)
DisjointObjectProperties(:subOrganizationOf :isPartOfUniversity)
DisjointObjectProperties(:subOrganizationOf :listedCourse)
DisjointObjectProperties(:subOrganizationOf :member)
DisjointObjectProperties(:subOrganizationOf :memberOf)
DisjointObjectProperties(:subOrganizationOf :orgPublication)
DisjointObjectProperties(:subOrganizationOf :publicationAuthor)
DisjointObjectProperties(:subOrganizationOf :publicationDate)
DisjointObjectProperties(:subOrganizationOf :publicationResearch)
DisjointObjectProperties(:subOrganizationOf :researchProject)
DisjointObjectProperties(:subOrganizationOf :softwareDocumentation)
DisjointObjectProperties(:subOrganizationOf :softwareVersion)
DisjointObjectProperties(:subOrganizationOf :takesCourse)
DisjointObjectProperties(:subOrganizationOf :teacherOf)
DisjointObjectProperties(:subOrganizationOf :teachingAssistantOf)
DisjointObjectProperties(:subOrganizationOf :tenured)
DisjointObjectProperties(:takesCourse :advisor)
DisjointObjectProperties(:takesCourse :affiliatedOrganizationOf)
DisjointObjectProperties(:takesCourse :degreeFrom)
DisjointObjectProperties(:takesCourse :hasAlumnus)
DisjointObjectProperties(:takesCourse :hasExamRecord)
DisjointObjectProperties(:takesCourse :hasFaculty)
DisjointObjectProperties(:takesCourse :isPartOfUniversity)
DisjointObjectProperties(:takesCourse :listedCourse)
DisjointObjectProperties(:takesCourse :member)
DisjointObjectProperties(:takesCourse :memberOf)
DisjointObjectProperties(:takesCourse :orgPublication)
DisjointObjectProperties(:takesCourse :publicationAuthor)
DisjointObjectProperties(:takesCourse :publicationDate)
DisjointObjectProperties(:takesCourse :publicationResearch)
DisjointObjectProperties(:takesCourse :researchProject)
DisjointObjectProperties(:takesCourse :softwareDocumentation)
DisjointObjectProperties(:takesCourse :softwareVersion)
DisjointObjectProperties(:takesCourse :subOrganizationOf)
DisjointObjectProperties(:takesCourse :teacherOf)
DisjointObjectProperties(:takesCourse :teachingAssistantOf)
DisjointObjectProperties(:takesCourse :tenured)
DisjointObjectProperties(:teacherOf :advisor)
DisjointObjectProperties(:teacherOf :affiliatedOrganizationOf)
DisjointObjectProperties(:teacherOf :degreeFrom)
DisjointObjectProperties(:teacherOf :hasAlumnus)
DisjointObjectProperties(:teacherOf :hasFaculty)
DisjointObjectProperties(:teacherOf :isPartOfUniversity)
DisjointObjectProperties(:teacherOf :listedCourse)
DisjointObjectProperties(:teacherOf :member)
DisjointObjectProperties(:teacherOf :memberOf)
DisjointObjectProperties(:teacherOf :orgPublication)
DisjointObjectProperties(:teacherOf :publicationAuthor)
DisjointObjectProperties(:teacherOf :publicationDate)
DisjointObjectProperties(:teacherOf :publicationResearch)
DisjointObjectProperties(:teacherOf :researchProject)
DisjointObjectProperties(:teacherOf :softwareDocumentation)
DisjointObjectProperties(:teacherOf :softwareVersion)
DisjointObjectProperties(:teacherOf :subOrganizationOf)
DisjointObjectProperties(:teacherOf :takesCourse)
DisjointObjectProperties(:teacherOf :teachingAssistantOf)
DisjointObjectProperties(:teacherOf :tenured)
DisjointObjectProperties(:teachingAssistantOf :advisor)
DisjointObjectProperties(:teachingAssistantOf :affiliatedOrganizationOf)
DisjointObjectProperties(:teachingAssistantOf :degreeFrom)
DisjointObjectProperties(:teachingAssistantOf :hasAlumnus)
DisjointObjectProperties(:teachingAssistantOf :hasExamRecord)
DisjointObjectProperties(:teachingAssistantOf :hasFaculty)
DisjointObjectProperties(:teachingAssistantOf :isPartOfUniversity)
DisjointObjectProperties(:teachingAssistantOf :listedCourse)
DisjointObjectProperties(:teachingAssistantOf :member)
DisjointObjectProperties(:teachingAssistantOf :memberOf)
DisjointObjectProperties(:teachingAssistantOf :orgPublication)
DisjointObjectProperties(:teachingAssistantOf :publicationAuthor)
DisjointObjectProperties(:teachingAssistantOf :publicationDate)
DisjointObjectProperties(:teachingAssistantOf :publicationResearch)
DisjointObjectProperties(:teachingAssistantOf :researchProject)
DisjointObjectProperties(:teachingAssistantOf :softwareDocumentation)
DisjointObjectProperties(:teachingAssistantOf :softwareVersion)
DisjointObjectProperties(:teachingAssistantOf :subOrganizationOf)
DisjointObjectProperties(:teachingAssistantOf :takesCourse)
DisjointObjectProperties(:teachingAssistantOf :teacherOf)
DisjointObjectProperties(:teachingAssistantOf :tenured)
DisjointObjectProperties(:tenured :advisor)
DisjointObjectProperties(:tenured :affiliatedOrganizationOf)
DisjointObjectProperties(:tenured :degreeFrom)
DisjointObjectProperties(:tenured :hasAlumnus)
DisjointObjectProperties(:tenured :hasExamRecord)
DisjointObjectProperties(:tenured :hasFaculty)
DisjointObjectProperties(:tenured :isPartOfUniversity)
DisjointObjectProperties(:tenured :listedCourse)
DisjointObjectProperties(:tenured :member)
DisjointObjectProperties(:tenured :memberOf)
DisjointObjectProperties(:tenured :orgPublication)
DisjointObjectProperties(:tenured :publicationAuthor)
DisjointObjectProperties(:tenured :publicationDate)
DisjointObjectProperties(:tenured :publicationResearch)
DisjointObjectProperties(:tenured :researchProject)
DisjointObjectProperties(:tenured :softwareDocumentation)
DisjointObjectProperties(:tenured :softwareVersion)
DisjointObjectProperties(:tenured :subOrganizationOf)
DisjointObjectProperties(:tenured :takesCourse)
DisjointObjectProperties(:tenured :teacherOf)
DisjointObjectProperties(:tenured :teachingAssistantOf)
```