Semantic Web   Autumn 2011   Coursework

For return on 20 January 2012 (late submission: 3 February 2012)

Electronic submission: **.pdf** *and .owl files only*

1. (8%) Consider the following RDF document:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
          xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
          xmlns:lit="http://literature.org/#"
          xml:base="http://literature.org/">

    <rdf:Description rdf:about="#Hamlet">
        <rdf:type rdf:resource="#drama"/>
    </rdf:Description>

    <rdf:Description rdf:about="#Sonet96">
        <rdf:type rdf:resource="#poem"/>
    </rdf:Description>

    <rdf:Description rdf:about="#wrote">
        <rdf:type rdf:resource="&rdf;Property"/>
        <rdf:domain rdf:resource="#writer"/>
        <rdf:range rdf:resource="#literary_content"/>
    </rdf:Description>

    <rdfs:Class rdf:about="#poem"/>
    <rdf:Class rdf:about="#drama"/>

    <lit:poet rdf:about="#Shakespear">
        <lit:wrote rdf:resource="#Sonet96"/>
        <lit:wrote rdf:resource="#Hamlet"/>
        <rdf:type rdf:resource="#playwright"/>
    </lit:poet>

</rdf:RDF>
```

(a) Describe in natural language the content of this document.
(b) Draw the graph representation of the document.
(c) Explain which of the following can or cannot be represented in RDF(S):

– Poets and playwrights are writers.

– Poets write poems; playwrights write dramas.

– Poets do not write RDF documents.

2. (10%) Represent the following information by means of RDF/S triples:

– a person, whose emails address is `bob@somewhere.uk`, plays golf with the spouse of a colleague who has two email addresses: `rob@elsewhere.uk` and `rob@somewhere.it`.

(Hint: use blank nodes as on pages 14–16, Lecture 3. You can use the Turtle syntax.) Draw the graph representation of the RDF/S document. Represent the same information using the language of description logic.

3. (10%) Consider the following data and background knowledge:

(1) John manages 'Eden'.

(2) Persons who manage projects are managers.

(3) One can only manage projects or departments.

(4) All managers manage something.

(5) Managers can be either top managers or area managers.

(6) Every employee has a manager and works on a project.

(7) One can only work on a project.

(8) Area managers manage projects.

(9) Those who have managers are employees.

(10) John is not a top manager.

(a) Which of the above statements can be satisfactorily modelled in RDF/S? Explain your answer and give the corresponding RDF/S graph representations.

(b) Represent the statements above as a description logic knowledge base, indicating the TBox and the ABox.

(c) What is the answer to the query 'is Eden a project?' with respect to the resulting knowledge base.

4. (16%) Sketch a normalised *movie ontology*, which covers the items listed below and provides sufficient concepts and roles for part (b).

(a) Indicate the hierarchies for both concepts (classes) and properties (roles). For concepts, indicate clearly which are self-standing, which are definable and which are modifiers. Introduce instances of concepts if required. For properties, define their domains and ranges, their inverses, and indicate whether the roles are (ir)reflexive, (a)symmetric, functional, inverse-functional or transitive.

`Items to be represented:` Genre, Award, Actor, Production Company, Action, Comedy, Director, Oscar, Hollywood.

(b) Define classes using your ontology for the items below (using the OWL functional-style syntax or a reasonable approximation) or explain why they cannot be expressed in OWL:

- Award winning comedy actor.
- Hollywood movie made in 1961.
- Company producing comedies and documentaries only.

If the definition in English is ambiguous, paraphrase it so it is unambiguous and then express the disambiguated notion in OWL.

5. (10%) Consider the following small ontology written in OWL functional-style syntax:

DisjointClasses(Animal Plant)
ObjectPropertyDomain(eats Animal)
EquivalentClasses(Herbivore ObjectAllValuesFrom(eats Plant))
EquivalentClasses(Carnivore ObjectAllValuesFrom(eats Animal))
EquivalentClasses(CarnivorousPlant ObjectIntersectionOf(Plant Carnivore))

(a) Write down DL equivalents of each of the axioms in the ontology.

(b) Enter the above ontology into Protégé, and use the reasoner to compute the class hierarchy.

(c) Explain why Plant is a sub-class of both Herbivore and Carnivore.

(d) This does not seem to be correct. Explain how you would improve the ontology in order to fix this problem.

(e) Did your improvement reveal any other problem with the ontology? If so, how would you repair the problem?

6. (10%) Consider the description logic knowledge base $\mathcal{K}$

$$\begin{aligned} \mathsf{Student} &\sqsubseteq \exists \mathsf{hasTutor}.\mathsf{Professor} \\ \mathsf{Professor} &\sqsubseteq (\exists \mathsf{teaches}.\mathsf{Student}) \sqcap (\neg \forall \mathsf{teaches}.\mathsf{Student}) \\ &\mathsf{joe} : \mathsf{Student} \\ &\mathsf{sam} : \mathsf{Student} \\ &(\mathsf{sam}, \mathsf{john}) : \mathsf{hasTutor} \end{aligned}$$

and the interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where

$$\begin{aligned} &\Delta^{\mathcal{I}} = \{a, b, c, d\}, \quad \mathsf{joe}^{\mathcal{I}} = a, \quad \mathsf{sam}^{\mathcal{I}} = b, \quad \mathsf{john}^{\mathcal{I}} = c, \\ &\mathsf{Student}^{\mathcal{I}} = \{a, b\}, \quad \mathsf{Professor}^{\mathcal{I}} = \{c, d\}, \\ &\mathsf{hasTutor}^{\mathcal{I}} = \{(a, c), (b, c)\}, \quad \mathsf{teaches}^{\mathcal{I}} = \{(c, b), (c, d), (d, b)\}. \end{aligned}$$

(a) Is $\mathcal{I}$ a model of $\mathcal{K}$? Explain your answer by computing the extensions $C^{\mathcal{I}}$ of all the concepts C from $\mathcal{K}$. Does $\mathcal{I}$ satisfy the unique name assumption?

(b) Is john an instance of Professor with respect to $\mathcal{K}$? Explain your answer.

(c) Is the concept $\mathsf{Student} \sqcap \exists \mathsf{hasTutor}.(\forall \mathsf{teaches}.\mathsf{Student})$ satisfiable with respect to $\mathcal{K}$? Explain your answer.

(d) Extend $\mathcal{K}$ with $\mathsf{Student} \equiv \mathsf{Professor}$. Is the resulting knowledge base consistent? Explain your answer.

(e) Assume that you have a reasoner that can check satisfiability of concepts with respect to a knowledge base. Can you use it to check that one concept is equivalent to another with respect to the knowledge base, for instance, whether $\mathsf{Student} \equiv \mathsf{Professor}$? Explain your answer.

7. (9%) Consider the following ABox $\mathcal{A}$:

$$\begin{array}{rr} (\mathsf{ron}, \mathsf{claudia}) : \mathsf{likes} & (\mathsf{ron}, \mathsf{peter}) : \mathsf{likes} \\ (\mathsf{claudia}, \mathsf{peter}) : \mathsf{is\_neighbour\_of} & (\mathsf{peter}, \mathsf{andrea}) : \mathsf{is\_neighbour\_of} \\ \mathsf{claudia} : \mathsf{Blond} & \mathsf{andrea} : \neg\mathsf{Blond} \end{array}$$

(a) Does $\mathcal{A}$ have a model?

(b) Is ron an instance of the concept

$$\exists \mathsf{likes}.(\mathsf{Blond} \sqcap \exists \mathsf{is\_neighbour\_of}.\neg\mathsf{Blond})$$

in all models of $\mathcal{A}$?

(c) Is ron an instance of the concept

$$\exists \mathsf{likes}.(\exists \mathsf{is\_neighbour\_of}.(\forall \mathsf{is\_neighbour\_of}.\neg\mathsf{Blond}))$$

in all models of $\mathcal{A}$?

8. (10%) Consider the concept

$$C \;=\; \neg(\forall R.\neg A \sqcup \forall R.\neg(B \sqcap D)) \sqcap \neg\exists R.(A \sqcap B \sqcap D)$$

(i) Transform concept C to an equivalent concept in negation normal form.

(ii) Prove that concept C is satisfiable by giving a model of C. To construct the model, apply the $\mathcal{ALC}$ tableau algorithm to the negation normal form of C obtained in (i).

9. (10%) Using the individuals john and helen, concept names PoorPerson, WealthyPerson, SmartPerson, PersonReadingNewspapers, HappyPerson, ExcitingSpouse and PersonWithExcitingLife and role hasSpouse, represent the following informal knowledge as an $\mathcal{ALC}$ knowledge base:

(1) People who are smart and not poor are happy.

(2) People who read newspapers are smart.

(3) John is wealthy.

(4) Helen reads newspapers and is wealthy.

(5) Happy people have exciting lives.

(6) Wealthy people are not poor.

(7) John has Helen as spouse.

(8) A spouse who is not exciting has only poor spouse.

(9) People who have a spouse with an exciting spouse are happy.

(a) Which of the statements above can be represented in RDF or RDFS?

(b) For each individual name, find the most specific (i.e., smallest) concepts containing it as an instance with respect to the knowledge base.

10. (3%) Explain how the notion of SymmetricProperty in OWL can be expressed in terms of other language primitives.

11. (4%) Write down one or more DL axioms that express the same constraints that are expressed in each of the following OWL axioms written in abstract syntax:

- DisjointClasses(Meat Fish Vegetable Fruit)
- ObjectPropertyDomain(father Person)
- ObjectPropertyRange(father Male)
- SubObjectPropertyOf(father parent)
- FunctionalObjectProperty(father)
- SymmetricObjectProperty(hasSameGrade)
- SubClassOf(Lion ObjectIntersectionOf(Animal ObjectAllValuesFrom(eats Animal)))
- EquivalentClasses(Carnivore ObjectIntersectionOf(Animal ObjectSomeValuesFrom(eats Animal)))