



MIDSUMMER EXAMINATIONS 2017

**DO NOT OPEN THE QUESTION PAPER UNTIL INSTRUCTED TO DO SO BY
THE CHIEF INVIGILATOR**

Department	Informatics
Module Code	CO7216
Module Title	Semantic Web
Exam Duration	Two Hours

CHECK YOU HAVE THE CORRECT QUESTION PAPER

Number of Pages	8
Number of Questions	2
Instructions to Candidates	Please do not remove this paper from the examination hall. This paper contains TWO questions. All questions carry equal weight (50 marks per question). Full marks may be obtained for answers to two questions. All relevant namespaces have been provided in the helper sheet.

FOR THIS EXAM YOU ARE ALLOWED TO USE THE FOLLOWING:

Calculators	No
Books/Statutes provided by the University	No
Are students permitted to bring their own Books/Statutes/Notes?	No
Additional Stationery	No



Dwelling Scenario

The Dwelling Scenario described below is used in parts of both Questions 1 and 2.

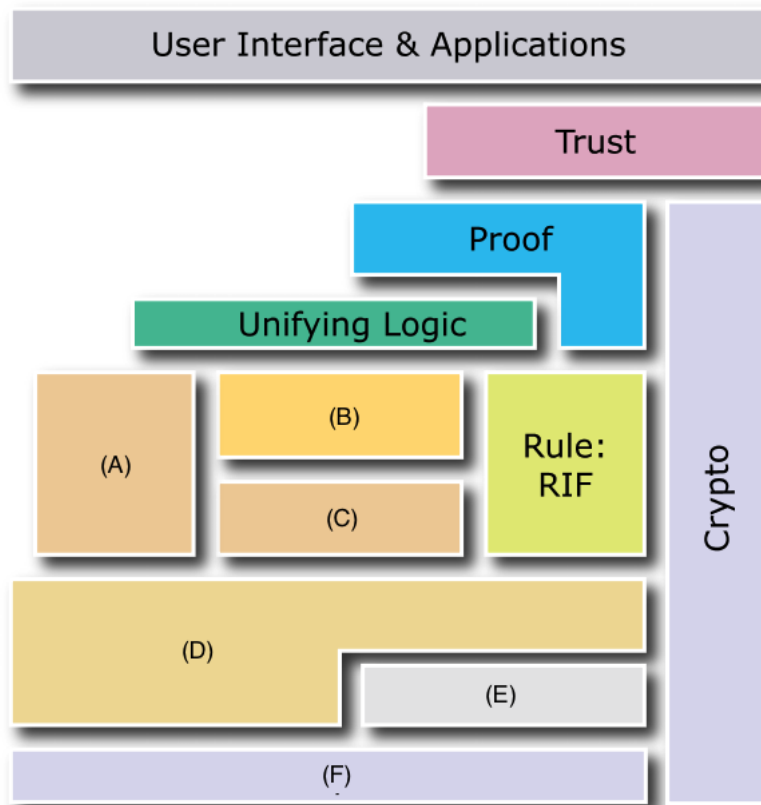
A dwelling is a self-contained unit of accommodation that functions as a home. Houses and flats are the most common types of dwelling in the UK.

In general, houses can be classified into these categories: bungalows, detached, semi-detached and terraced houses. A bungalow is a house with only a ground floor and no stairs; a detached house is a house that does not share party walls with any other dwellings (a party wall is a dividing partition between two adjoining buildings); a semi-detached house is one that is joined to another similar house on only one side; a terraced house is one of a row of similar houses joined together by their party walls. Flats are usually parts of a block with shared communal areas such as landings, stairwells and walkways. A studio flat combines living room, bedroom, and kitchenette into a single room; Halls of residence are purpose-built flats only for students.

House or flats may contain different types of room such as bedroom, kitchen or bathroom. The conventional way of categorising dwellings is to look at the number of bedrooms. For example, a 3-bedroom house has three bedrooms. A bathroom is a room with a toilet, sink and either a shower or bathtub. An en-suite bedroom is a bedroom that has a bathroom directly attached to it. A family bathroom is a bathroom not attached to a bedroom, but with its door opening onto e.g. a corridor.

**Question 1.**

1. (a) Name the missing components (A), (B), (C), (D), (E) and (F) in the Semantic Web Layer Cake below: [3 marks]



- (b) Consider the XML document `jobs.xml` in **Appendix A**.

- Write an RDF instance document corresponding to lines 24-35 of `jobs.xml` using RDF/XML syntax. [5 marks]
- Write an RDF Schema document for the RDF instance document in (i) using Turtle Notation. [10 marks]
- For each question below, write a SPARQL query to get the specified result:
 - Select anyone applying for Graphic designer, who knows Photoshop and CSS, and has at least two years of experience. Limit the result to the first 2 rows.
 - Get a list of applicants with a master's degree (MA or MSc), displaying their employment histories (if given).
 - Show the total number of technical skills in each skill sub-category (e.g. language, framework and tool) among all candidates, displaying the results in ascending order.

[9 marks]



- (c) Draw an RDF graph corresponding to the following Turtle Notation: [4 marks]

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
[foaf:age "25" ] foaf:knows
[ foaf:knows [foaf:nick "Jimmy" ] ;
foaf:based_near <http://dbpedia.org/ontology/Leicester> ;
foaf:homepage <http://www.example.com/document/myhome> ] .
```

- (d) Explain the differences between the RDF(S) core elements (or attributes) below:

- i. `rdf:ID` and `rdf:about`
- ii. `rdfs:domain` and `rdfs:range`
- iii. `rdfs:Class` and `rdf:type`
- iv. `rdf:predicate` and `rdf:Property`

[4 marks]

- (e) Suppose `C` is a class in the ontology. Write a SPARQL query and any necessary reasoning rules to obtain its direct and indirect instances (instances of `C` and instances of its subclasses).

[5 marks]

- (f) Consider the dwelling scenario.

- Identify four concepts and four properties relevant to the domain. You can write your answer as:

Concepts: `Concept1`, `Concept2`, ...

Properties: `Property1`, `Property2`, ...

- Define the hierarchies (subclasses) for at least three top-level concepts in this domain. The ontology should demonstrate the use of multiple inheritance. Give an example of the property-subproperty relation between two properties. You may draw the hierarchies as diagrams or write the ontology in RDF/XML or Turtle syntax.

Note: you may add other concepts and it may be necessary to do so to complete this question.

[10 marks]

**Question 2.**

2. (a) Which is the most appropriate OWL 2 profile if you want to perform scalable reasoning without sacrificing too much expressive power? Give an example of an OWL 2 construct you should avoid when choosing this profile. [3 marks]
- (b) Explain the differences between `owl:intersectionOf` and `owl:Restriction` with a diagram. Use examples from the Dwellings scenario to explain each concept. [6 marks]
- (c) Which of the following statements are true? For each of the false statements, justify your answer.
- i. All Functional Properties are Object Properties.
 - ii. An OWL individual cannot have two disjoint property values.
 - iii. `owl:inverseFunctionalProperty` is a subclass of `owl:inverseOf` and `owl:FunctionalProperty`.
 - iv. RDF Schema does not support cardinality restrictions.
 - v. A Symmetric or Transitive Property must be an Object Property. `owl:sameIndividualAs` is a Symmetric, Reflexive and Transitive Property.
 - vi. `owl:differentFrom` states that a collection of individuals is pairwise distinct.
 - vii. A Description Logic based reasoner can be used to check the consistency of an OWL Full ontology.
- [7 marks]
- (d) For each statement below, write a reasoning rule using SWRL or Jena rule syntax:
- i. A Nephew is the son of one's brother or sister. (Note: use only `hasSibling(?,?)`, `hasChild(?,?)`, `Male(?,?)` and `Nephew(?,?)` to express this rule).
 - ii. If an OWL property is linked to `owl:TransitiveProperty` via `rdf:type` then this property is transitive.
 - iii. Given an RDF triple statement $\langle s, p, o \rangle$, if the range of the property (predicate) p is class C then o must be an instance of C .

[6 marks]



- (e) Use RDFa or Microdata to annotate the following HTML snippet (See **Appendix B** for more information). [4 marks]

```
<div>
  <p>Lord Richard Attenborough played the character
    of John Hammond in the 1993 film Jurassic Park.
  </p>
</div>
```

- (f) Within the context of linked data, discuss the advantages and disadvantages of two different strategies for dereferencing URIs. [4 marks]
- (g) Consider the dwelling scenario. Write an OWL class expression in RDF/XML or OWL Manchester syntax that corresponds to the following OWL Functional syntax class expression. [4 marks]

```
Declaration(Class(:EnsuiteBedroom))
EquivalentClasses(:EnsuiteBedroom
ObjectIntersectionOf(
  ObjectSomeValuesFrom(:connectsTo :Bathroom) :Bedroom))
SubClassOf(:EnsuiteBedroom :Bedroom)
```

- (h) Consider the dwelling scenario. As part of the ontology definition for the scenario, define OWL classes for the entities in (i), (ii) and (iii) below. Use OWL Manchester syntax, OWL Functional syntax or RDF/XML as you wish. Assume all definitions are in the default namespace.
- i. Define an entity demonstrating the use of `owl:allValuesFrom`. [2 marks]
 - ii. Define an entity demonstrating the construction of an OWL class by using `owl:cardinality`, `owl:complementOf` and `owl:someValuesFrom`. [4 marks]
 - iii. Define an OWL class for four-bedroom house with at least one family bathroom. [6 marks]
- (i) Consider the dwelling scenario. Give four examples of properties specific to the domain. Properties types may be chosen from transitive, symmetric, asymmetric, functional, inverseFunctional, reflexive or irreflexive respectively. Do not give more than one example of each property type. Define the domain and range restrictions for the properties. [4 marks]



Appendix A: jobs.xml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <candidates>
3   <candidate id="001" name="Thomas">
4     <position>Web Designer</position>
5     <years_of_experience>1</years_of_experience>
6     <education>BSc in Computing</education>
7     <technical_skills>
8       <language>JavaScript</language>
9       <framework>jQuery</framework>
10      <tool>Dreamweaver</tool>
11      <language>HTML</language>
12      <language>CSS</language>
13    </technical_skills>
14  </candidate>
15  <candidate id="002" name="Phoebe">
16    <position>Graphic Designer</position>
17    <years_of_experience>2</years_of_experience>
18    <education>MA in Creative Arts and Design</education>
19    <technical_skills>
20      <tool>Photoshop</tool>
21      <language>CSS</language>
22    </technical_skills>
23  </candidate>
24  <candidate id="003" name="Bob">
25    <position>Programmer</position>
26    <years_of_experience>4</years_of_experience>
27    <education>MSc in Advanced Software Engineering</education>
28    <previous_employer>Microsoft</previous_employer>
29    <technical_skills>
30      <language>Java</language>
31      <language>C#</language>
32      <framework>SpringMVC</framework>
33      <language>HTML</language>
34    </technical_skills>
35  </candidate>
36  <candidate id="004" name="Alice">
37    <position>Programmer</position>
38    <years_of_experience>1</years_of_experience>
39    <education>BSc in Computer Science</education>
40    <technical_skills>
41      <language>Java</language>
42      <language>JavaScript</language>
43      <framework>jQuery</framework>
44      <language>HTML</language>
45    </technical_skills>
46  </candidate>
47 </candidates>
```



Appendix B

(a) Namespaces

```
base="http://www.cs.le.ac.uk/co7216/exam/#"  
rdfs="http://www.w3.org/2000/01/rdf-schema#"  
owl="http://www.w3.org/2002/07/owl#"  
xsd="http://www.w3.org/2001/XMLSchema#"  
rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"  
foaf="http://xmlns.com/foaf/0.1/"  
dc="http://purl.org/dc/elements/1.1/"  
po="http://purl.org/ontology/po"
```

Default namespace:

```
http://www.cs.le.ac.uk/co7216/exam/#
```

(b) Classes and properties for Q2.e.

```
classes: foaf:Person, dbo:Film; properties: foaf:name, dbo:starring
```