Question Paper

Exam Date & Time: 08-Jan-2024 (02:30 PM - 05:30 PM)

MANIPAL ACADEMY OF HIGHER EDUCATION

SEVENTH SEMESTER B.TECH END SEMESTER EXAMINATIONS, JAN 2024

Semantic Web [ICT 4036]

Marks: 50 **Duration: 180 mins.**

A

Answer all the questions.

Instructions to Candidates: Answer ALL questions Missing data may be suitably assumed

1) A) Consider a taxonomy where 'countries' is a class, and 'cities' is a subclass. Introduce a property 'population' with a range of integers. Suggest a relevant constraint using OWL for the 'population' property within the 'large cities' subclass. (Large city has minimum 10000 and maximum 1000000 population). Represent the same in RDF/XML syntax. Discuss the feasibility of expressing this constraint in RDF Schema, possibly utilizing rdfs:range, and provide a justification for your stance. (5)

B) Consider the following SPARQL query: (3)

```
SELECT ?book ?language1 ?language2
WHERE {
 ?book :translatedInto ?language1 .
 ?book :translatedInto ?language2 .
 FILTER (?language1 != ?language2) .
 FILTER (COUNT(?language) >= 2) . }
```

1. What is the interpretation of this query?
2. Draw the RDF graph corresponding to the graph patterns of the query.

C) Construct and add the following relation using SPARQL SPIN *uncle(X,Z) ← parent(Y,Z), parent(X, Y), sibling(X,W), male(W).* (2)

2) A) Consider the RDF triples shown in the Figure. Write SPARQL queries to (5)

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix foaf: <http://xmlns.com/foaf/0.1/>.
@prefix ex: <http://example.org/>.

ex:person1 rdf:type foaf:Person;
      foaf:name "Alice";
      foaf:email "alice@example.com";
      ex:hasInterest ex:Programming.

ex:person6 rdf:type foaf:Person;
      foaf:name "Frank";
      ex:hasInterest ex:Programming;
      ex:hasFriend ex:person5.
```

```
ex:person2 rdf:type foaf:Person;
      foaf:name "Bob";
      ex:hasInterest ex:Programming.

ex:person3 rdf:type foaf:Person;
      foaf:name "Charlie";
      ex:hasInterest ex:Art;
      ex:hasFriend ex:person1.

ex:person4 rdf:type foaf:Person;
      foaf:name "David";
      ex:hasInterest ex:Programming;
      ex:hasFriend ex:person2;
      foaf:age 35.

ex:person5 rdf:type foaf:Person;
      foaf:name "Eva";
      ex:hasInterest ex:Art.

ex:person7 rdf:type foaf:Person;
      foaf:name "Grace";
      ex:hasInterest ex:Art;
      ex:hasFriend ex:person6.

ex:person8 rdf:type foaf:Person;
      foaf:name "Harry";
      ex:hasInterest ex:Programming;
      ex:hasFriend ex:person7;
      foaf:age 28.

ex:person9 rdf:type foaf:Person;
      foaf:name "Ivy";
      ex:hasInterest ex:Art;
      ex:hasFriend ex:person8.

ex:Programming rdf:type ex:Interest.

ex:Art rdf:type ex:Interest.
```

a) Retrieve all individuals and their interests.

b) Find individuals with no friends older than 30.

c) List all books and their authors along with available genres (if any).

B) Represent the Subclasses and Subproperties axiomatically using basic predicate logic along with suitable examples. (3)

C) Write a sparql query to check if there are any individual individuals interested in Programming. Consider the owl file structure shown in figure (2)

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix foaf: <http://xmlns.com/foaf/0.1/>.
@prefix ex: <http://example.org/>.

ex:Book rdf:type rdfs:Class.
ex:Author rdf:type rdfs:Class.
ex:Genre rdf:type rdfs:Class.
ex:hasAuthor rdf:type rdf:Property.
ex:hasGenre rdf:type rdf:Property.

ex:book1 rdf:type ex:Book;
      ex:bookTitle "The Catcher in the
      ex:hasAuthor ex:author1;
      ex:hasGenre ex:Fiction.

      ex:bookTitle "Introduction to Algorithms";
      ex:hasAuthor ex:author3;
      ex:hasGenre ex:NonFiction.

ex:author1 rdf:type ex:Author;
      ex:authorName "J.D. Salinger".

ex:author2 rdf:type ex:Author;
      ex:authorName "Harper Lee".
```

```
ex:book2 rdf:type ex:Book;
    ex:bookTitle "To Kill a Mockingbir
    ex:hasAuthor ex:author2;
    ex:hasGenre ex:Fiction.

ex:author3 rdf:type ex:Author;
    ex:authorName "Thomas H. Cormen".

ex:Fiction rdf:type ex:Genre.
ex:NonFiction rdf:type ex:Genre.
```

a. Write a SPARQL query to check if there are triples representing books in the Science Fiction genre.

b. Write a SPARQL query to create a new graph with books and their authors in the Science Fiction genre.

3) Description logic and Horn Logic are orthogonal to each other. Justify the statement with respect expressiveness of each with supporting examples. (5)

A)

B) Following are the Horn Logic representations of OWL concept, where C, D are classes and P represents property. Analyse and interpret the concepts represented by the following? (3)

a. **C(X) → D(X) , D(X) → C(X)**

b. **C1(X), C2 (X) → D(X)**

c. **C(X),P(X,Y) → D(Y)**

C) Represent the information in the table in RDF?XML and Turtle template considering Artist as subject. (2)

Title	Artist	Country	Company	Price	Year
Empire Burlesque	Bob Dylan	USA	Columbia	10.90	1985
Hide your heart	Bonnie Tyler	UK	CBS Records	9.90	1988

4) Illustrate Rule Interchange Format categories of OWL2 RL rules. (5)

A)

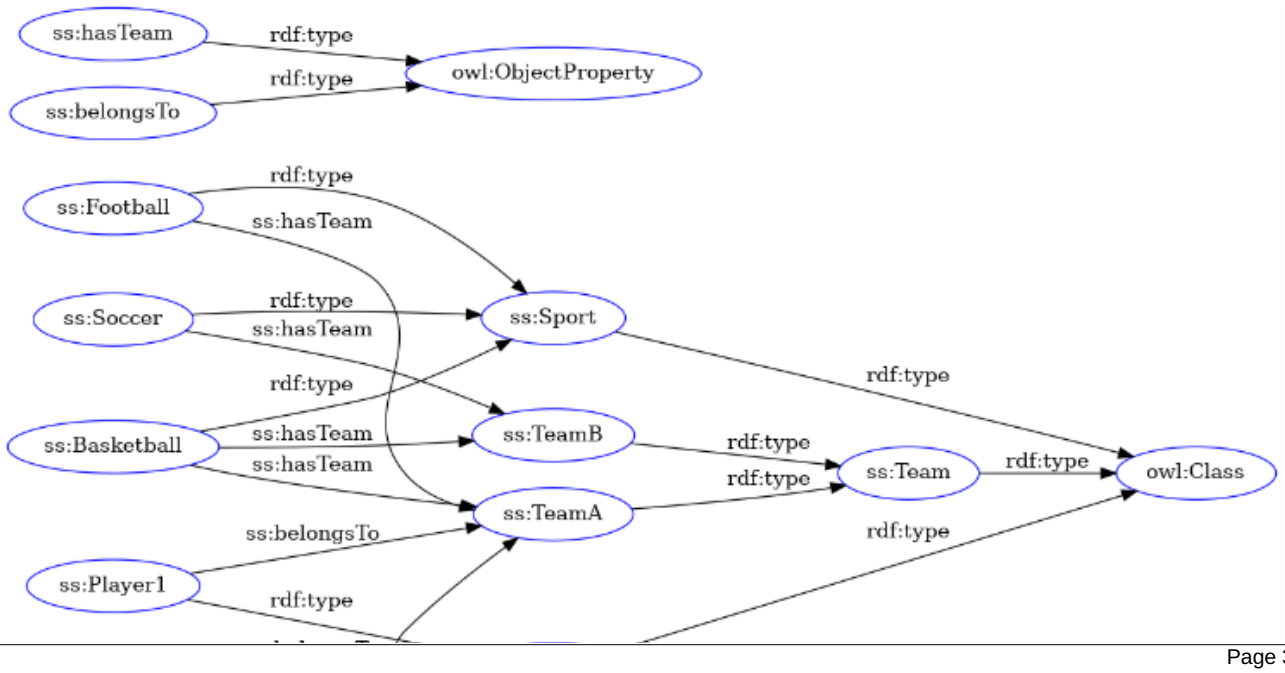
B) Represent the following rule in SPARQL SPIN: C2(X) ← C1(X), equivalentClass(C1, C2). (3)

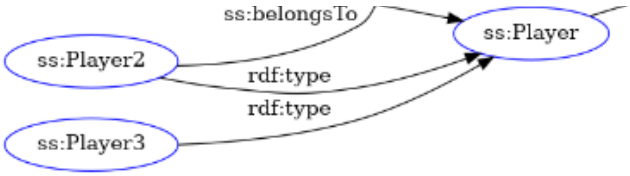
C) Differentiate existential and Universal Restriction axioms with examples. (2)

5) In many application multiple ontologies have to be combined. In this process challenge is ontology mapping. How is consistent mapping achieved currently? (5)

A)

B) From the RDF graph given, infer all the RDF triple for each subject associated with TeamA. Also represent the triples in Turtle syntax, along with proper prefixes. (3)





Namespaces:
rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs: http://www.w3.org/2000/01/rdf-schema#
owl: http://www.w3.org/2002/07/owl#
ss: http://example.org/healthcare#

C) How to represent anonymous resources in RDF/XML and Turtle serialization syntaxes. What do anonymous resource indicate?. (2)

-----End-----