

Lab 7 – OWL in Space

CC7220-1 – September 30, 2024

We will use OWL to define some semantics for astronomic bodies. Visit <http://rdfplayground.dcc.uchile.cl>. Copy and paste the following data into the text field on the left side of the page:

```
@prefix ex: <http://ex.org/>.
@prefix owl: <http://www.w3.org/2002/07/owl#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.

ex:Sol a ex:LoneStar .
ex:Mercury a ex:Planet ; ex:hasParent ex:Sol .
ex:Venus a ex:Planet ; ex:hasParent ex:Sol .
ex:Earth a ex:Planet ; ex:hasParent ex:Sol ; ex:hasChild ex:Luna .
ex:Mars a ex:Planet ; ex:hasParent ex:Sol ; ex:hasChild ex:Phobos .
ex:Jupiter a ex:Planet ; ex:hasParent ex:Sol ; ex:hasChild ex:Ganymede .
ex:Saturn a ex:Planet ; ex:hasParent ex:Sol ; ex:hasChild ex:Titan .
ex:Uranus a ex:Planet ; ex:hasParent ex:Sol ; ex:hasChild ex:Titania .
ex:Neptune a ex:Planet ; ex:hasParent ex:Sol ; ex:hasChild ex:Triton .
ex:Pluto a ex:DwarfPlanet ; ex:hasParent ex:Sol ; ex:hasChild ex:Charon .
ex:Charon a ex:SolarPlanemo ; ex:hasParent ex:Sol .
ex:Ceres a ex:DwarfPlanet ; a ex:SolarBody ; ex:hasStar ex:Sun .

ex:Phobos ex:discoveredBy ex:AsaphHall . ex:Ganymede ex:discoveredBy ex:GalileoGalilei .
ex:Titan ex:discoveredBy ex:ChristiaanHuygens . ex:Triton ex:discoveredBy ex:WilliamLassell .
ex:Titania ex:discoveredBy ex:WilliamHerschel . ex:Charon ex:discoveredBy ex:JamesWChristy .

ex:AlphaCentauriA a ex:TripleStar ; ex:hasSibling ex:AlphaCentauriB , ex:AlphaCentauriC , ex:ProximaCentauri .
ex:AlphaCentauriB a ex:YellowDwarfStar . ex:AlphaCentauriC a ex:RedDwarfStar .
ex:ProximaCentauri a ex:RedDwarfStar .

ex:hasParent owl:inverseOf ex:hasChild ; rdfs:subPropertyOf ex:hasAntecedent .
ex:hasStar rdfs:range ex:Star ; rdfs:subPropertyOf ex:hasAntecedent .
ex:hasAntecedent a owl:TransitiveProperty . ex:discoveredBy owl:inverseOf ex:discovered .
ex:hasSibling a owl:SymmetricProperty . ex:TripleStar rdfs:subClassOf ex:Star .
ex:LoneStar rdfs:subClassOf ex:Star .
```

The process for reasoning with be the same as for Lab 4.

On the right-hand side, you should add RDFS/OWL definitions and axioms that infer what is specified in the question (note that you cannot add the required data explicitly; it must be inferred through the requested RDF-S/OWL definitions). The definitions you add should accumulate. Note that you can define nested definitions as follows, which will count as one axiom (it states that: the class of *DCC students who are also masters or PhD students* is a sub-class-of the class of *entities that have a supervisor who is a professor on the DCC staff*):

```
[ owl:intersectionOf ( :DCCStudent [ owl:unionOf ( :MastersStudent :PhDStudent ) ] ) ] rdfs:subClassOf
[ owl:someValuesFrom [ owl:intersectionOf ( :Professor :DCCStaff ) ] ; owl:onProperty :supervisor ] .
```

In this (rather complex) example we used `rdfs:subClassOf` (rather than `owl:equivalentClass`) since other types of students may have DCC professors as supervisors.

As another example relevant to the data given, you can start with the following definition in the OWL panel, keeping the prefixes specified:

```
[ owl:allValuesFrom ex:TripleStar ; owl:onProperty ex:hasSibling ] owl:equivalentClass
[ owl:intersectionOf ( [ owl:someValuesFrom owl:Thing ; owl:onProperty ex:hasSibling ] ex:TripleStar ) ] .
```

which, selecting the OWL reasoner, should infer:

```
ex:AlphaCentauriB a ex:TripleStar . ex:AlphaCentauriC a ex:TripleStar .  
ex:ProximaCentauri a ex:TripleStar .
```

You might see that these entities are also instances of blank nodes; these refer to class definitions that the entity is an instance of, like entities that have some sibling, entities whose siblings are triple stars, etc.

Extend this example with RDFS/OWL axioms to answer the following:

1. [6 MARKS] Add *one* axiom to state that a Planemo is either a Planet or a DwarfPlanet (or both), and that all Planets and DwarfPlanets are Planemos, inferring that:

```
ex:Mercury a ex:Planemo . ex:Venus a ex:Planemo . ... ex:Ceres a ex:Planemo .
```

2. [6 MARKS] Add *one* axiom to state that all children of Planemos must be of type Moon, inferring that:

```
ex:Luna a ex:Moon . ex:Phobos a ex:Moon . ... ex:Charon a ex:Moon .
```

3. [6 MARKS] Add *one* axiom to state that all Moons, Planemos and Stars are of type AstronomicBody (but there may be other kinds of AstronomicBody), inferring that:

```
ex:Sol a ex:AstronomicBody . ex:Mercury a ex:AstronomicBody . ... ex:Triton a ex:AstronomicBody .
```

4. [6 MARKS] Add *one* axiom to state that any entity with antecedent Sol must be a SolarBody, and any SolarBody must have antecedent Sol, inferring that:

```
ex:Mercury a ex:SolarBody . ... ex:Triton a ex:SolarBody . ex:Ceres ex:hasAntecedent ex:Sol .
```

5. [6 MARKS] Add *one* axiom to state that an entity that is a SolarBody and a Planemo is a SolarPlanemo, and that every SolarPlanemo is a SolarBody and a Planemo, allowing to infer:

```
ex:Mercury a ex:SolarPlanemo . ... ex:Ceres a ex:SolarPlanemo . ex:Charon a ex:Planemo .
```

6. [6 MARKS] Add *one* axiom to state that an entity cannot be a Planemo and a Moon (making Charon inconsistent).

7. [6 MARKS] Add *one* axiom to state that an entity who has discovered an AstronomicBody must be a Person and have occupation Astronomer (but not vice-versa: not all astronomers discover astronomic bodies), inferring:

```
ex:AsaphHall a ex:Person ; ex:occupation ex:Astronomer . ...
```

8. [6 MARKS] Add *one* axiom to state that a star has *at most* zero children of type Moon, identifying Sol/Charon as inconsistent (note: *the reasoner does not support exact cardinalities, hence the "at most" is important*).

9. [6 MARKS] Add *one* axiom to state that an entity that has an antecedent that is a LoneStar has *at most* one antecedent that is a Star. This should infer (through Ceres):

```
ex:Sun owl:sameAs ex:Sol . ex:Mercury ex:hasParent ex:Sun . ... ex:Ceres ex:hasStar ex:Sol . ...
```

10. [6 MARKS] Add *one* axiom to state that an entity cannot be both a RedDwarfStar and a YellowDwarfStar. Add another axiom to define that a TripleStar has precisely two siblings (but not vice-versa; something with two siblings may not be a TripleStar). Indicate as a comment in your answer what inference we expect to find. (Unfortunately it will not be inferred because the reasoner is *incomplete*; more on this in the next class.)

SUBMIT: a single file lab7.ttl containing the RDFS/OWL axioms required to answer each question. Upload the *input* data only, not the output of the reasoner. You do not need to upload the data given in this document. Clearly indicate the triples added for each question with a preceding comment line (e.g., "# Q1" without the quotes).