Lab 8 – Zebroids, Zorses, Hebras, Zonkeys and OWL

CC7220-1 – October 7, 2024

For the final boss of OWL, we will look at more complex entailments involving existentials, disjunctions and counting. For this we will use a reasoner called HermiT[1]: an OWL 2 DL reasoner based on a tableau rather than rules. HerMiT will find all possible entailments and will halt on any valid input, but (unlike the previous rule-based reasoner) may reject ontologies with features that may lead to undecidability.

Unfortunately we have not quite yet managed to integrate HermiT with RDF Playground, but if you navigate to `http://cc7220.dcc.uchile.cl/lab06/`, you will find a simple interface with the following default data loaded for you:

```
@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#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.

ex:hasSire a owl:ObjectProperty .
ex:hasDam a owl:ObjectProperty .
ex:hasTwin a owl:ObjectProperty .

ex:Zeus a ex:Zonkey ; ex:hasTwin ex:Zev .

ex:Zach a ex:Zebroid ; ex:hasDam ex:Lea ; ex:hasSire ex:Marty .
ex:Zia ex:hasDam ex:Lea ; ex:hasSire ex:Marty .

ex:Zeb a ex:Zorse ; ex:hasDam ex:Lea ; ex:hasSire ex:Zamba .
ex:Zab ex:hasDam ex:Lea ; ex:hasSire ex:Zamba .

ex:Trip a ex:Horse ; ex:hasDam ex:Canela ; ex:hasSire ex:Jupiter .
ex:Hannah ex:hasDam ex:Zeta ; ex:hasSire ex:Jupiter .
ex:Zeta a ex:Zebra .

ex:hasDam rdfs:subPropertyOf ex:hasParent .
ex:hasSire rdfs:subPropertyOf ex:hasParent .
ex:hasSire owl:propertyDisjointWith ex:hasDam .

ex:Zebra rdfs:subClassOf ex:Equine .
ex:Donkey rdfs:subClassOf ex:Equine .
ex:Horse rdfs:subClassOf ex:Equine .
ex:Zebroid rdfs:subClassOf ex:Equine .
ex:Zorse rdfs:subClassOf ex:Equine .
ex:Hebra rdfs:subClassOf ex:Equine .
ex:Zonkey rdfs:subClassOf ex:Equine .

[ a owl:AllDisjointClasses ;
  owl:members ( ex:Zebroid ex:Zebra ex:Donkey ex:Horse ) ] .
```

Dam:Lea Zebra Sire:Marty non-zebra Equine

Zebroid

Zamba is a Zebra Lea is a HorseDam

dam:Zeta Zebra sire Jupiter Horse

You can simply write your solutions at the bottom of the data (there is only one input form this time).

Note that `ex:hasDam` denotes an equine mother and `ex:hasSire` an equine father. The final axiom denotes that the listed classes are pairwise disjoint. To these data, you should add RDFS/OWL definitions and axioms that infer the given data. Other intuitively true inferences not listed may also arise. Your answers should not mention any specific individual like `ex:Zach`, `ex:Lea`, etc., but rather should only define axioms on classes and properties. **For each question you may add one or more axioms; you may have to add axioms not described by the question**

[1] `http://www.hermit-reasoner.com/`

(a) Zorse

(b) Hebra

(c) Zonkey

Figure 1: Zebroids

but that intuitively reflect reality (e.g., defining that all equines have exactly two parents reflects reality but defining that they have only one parent does not reflect reality). Axioms should accumulate from question to question. As per previous labs, it is not permitted to add the intended inferences directly to the data.

You can invent new classes or new properties, but **this reasoner requires that classes** (that do not have explicit instances in the data) **are declared as such**, and that **properties are declared as datatype properties** (taking literal values), **or object properties** (taking IRI or blank node values). So (hypothetically) if you wished to add a class such as `ex:NonZebraEquine` and a new (object) property such as `ex:child`, you would need to add:

```
ex:NonZebraEquine a owl:Class .
ex:child a owl:ObjectProperty .
```

Let's start!

1. [15 MARKS] Noting that the twin of a *zonkey* is a *zonkey*, add axioms to infer that:

```
ex:Zev a ex:Zonkey .
```

2. [15 MARKS] Noting that a *zebroid* is any child of a zebra and a non-zebra equine, add axioms to infer that:

```
ex:Zia a ex:Zebroid .
```

3. [15 MARKS] Noting that a *zorse* is any child of a zebra sire and a horse dam, add axioms to infer that:

```
ex:Zab a ex:Zorse .
ex:Zach a ex:Zorse .
ex:Zia a ex:Zorse .
```

4. [15 MARKS] Noting that a *hebra* is any child of a zebra dam and a horse sire, add axioms to infer that:

```
ex:Hannah a ex:Hebra .
```

If you look through the results, you'll find some other interesting conclusions: that Marty is a zebra, that zorse and hebra are both sub-classes of zebroid, etc. Note that if you try your solutions in RDF Playground, you will be missing inferences! This is because the (OWL 2 RL/RDF) rules that RDF Playground uses are incomplete.

SUBMIT: a single file `lab8.ttl` containing the default data and all of the additional RDFS/OWL axioms required to answer each question. Ensure to upload the *input* data only, not the output of the reasoner. Clearly indicate the triples added for each question with a preceding comment line (e.g., `"# Q1"` without the quotes).