Lab 4 – SPARQLing Particles

CC7220-1 – August 25, 2025

Today we will query an RDF graph about the Standard Model of Particle Physics using SPARQL. We will use RDF Playground (`http://rdfplayground.dcc.uchile.cl/`). Copy and paste the following RDF graph describing sub-atomic particles into the text field on the left-hand side and view it as a graph.

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

:Up a :Quark ; :spin 0.5 ; :generation 1 ; :interaction :Strong , :EM , :Weak .
:Down a :Quark ; :spin 0.5 ; :generation 1 ; :interaction :Strong , :EM , :Weak .
:Charm a :Quark ; :spin 0.5 ; :generation 2 ; :interaction :Strong , :EM , :Weak .
:Strange a :Quark ; :spin 0.5 ; :generation 2 ; :interaction :Strong , :EM , :Weak .
:Top a :Quark ; :spin 0.5 ; :generation 3 ; :interaction :Strong , :EM , :Weak .
:Bottom a :Quark ; :spin 0.5 ; :generation 3 ; :interaction :Strong , :EM , :Weak .
:Electron a :Lepton ; :spin 0.5 ; :generation 1 ; :interaction :Weak .
:ElectronNeutrino a :Lepton ; :spin 0.5 ; :generation 1 ; :interaction :Weak .
:Muon a :Lepton ; :spin 0.5 ; :generation 2 ; :interaction :Weak .
:MuonNeutrino a :Lepton ; :spin 0.5 ; :generation 2 ; :interaction :Weak .
:Tau a :Lepton ; :spin 0.5 ; :generation 3 ; :interaction :Weak .
:TauNeutrino a :Lepton ; :spin 0.5 ; :generation 3 ; :interaction :Weak .
:Gluon a :GaugeBoson ; :spin 1 ; :mediates :Strong .
:Photon a :GaugeBoson ; :spin 1 ; :mediates :EM  .
:ZBoson a :GaugeBoson ; :spin 1 ; :mediates :Weak  .
:WBoson a :GaugeBoson ; :spin 1 ; :mediates :Weak  .
:Higgs a :ScalarBoson ; :spin 0 .

:Neutron a :Baryon ;
  :contains [ :component :Up ; :quantity 1 ] , [ :component :Down ; :quantity 2 ] .
:Proton a :Baryon ;
  :contains [ :component :Up ; :quantity 2 ] , [ :component :Down ; :quantity 1 ] .
```

You can write queries using the SPARQL tab on the right-hand side. When you write a query select the operation type (unless otherwise stated use `SELECT` queries) and whatever result format you prefer (*Text* is best for `SELECT` and `ASK`, while *TTL* is best for `CONSTRUCT`). To test it out, you can try this SPARQL query, which returns the components of each baryon:

```
PREFIX :  <http://ex.org/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT ?baryon ?comp WHERE {
        ?baryon a :Baryon ; :contains ?blank .
        ?blank :component ?comp .
}
```

The queries should not introduce more information than in the question; for example, if the question asks for the force mediated by gluons, you cannot simply insert `:Strong` in the query, but rather the query should find `:Strong`. Note that the term "particle" is used like "resource" or "entity" in order to phrase the questions more naturally; you do not need to explicitly check that something is of type particle in the query (though for more specific types like leptons, muons, etc., you should check the type). You can return whatever variables you feel are relevant for the query. Remove duplicate results where present. *Let's begin!*

1. [6 MARKS] Find the spin of a photon.

2. [6 MARKS] Find particles that mediate the weak force and their spin.
3. [6 MARKS] Find particles that interact through the force mediated by gluons.
4. [6 MARKS] Find the forces through which each baryon interacts based on its components.
5. [6 MARKS] Find particles whose spin is a whole number.
6. [6 MARKS] Find particles that interact through the force mediated by gluons or photons.
7. [6 MARKS] Find particles with positive spin (>0) and, if given, their generation (if not, still return the result).
8. [6 MARKS] Find particles that interact through the weak force but not the strong force.
9. [6 MARKS] Find if there are particles with lower (<) spin than a muon (use `ASK`).
10. [6 MARKS] Create the following graph connecting quarks in the next or previous generation (use `CONSTRUCT`).

```
@prefix : <http://ex.org/>

:Up :nextGeneration :Charm , :Strange .
:Down :nextGeneration :Charm , :Strange .
:Charm :nextGeneration :Top , :Bottom ; :prevGeneration :Up , :Down .
:Strange :nextGeneration :Top , :Bottom ; :prevGeneration :Up , :Down .
:Top :prevGeneration :Charm , :Strange .
:Bottom :prevGeneration :Charm , :Strange .
```

Note that for question 10, since the output is an RDF graph, you can choose to visualise the results in RDF Playground as a graph.

SUBMIT: A single file `lab4.sparql` containing the query required to answer each question. Ensure to upload the *queries* only, not the results. Indicate the question to which each query responds with a preceding comment (e.g., `"# Q1"` without the quotes).