

Knowledge Graphs

Exercise Sheet 2: Graphs and RDF

Maximilian Marx, Markus Krötzsch
2026-11-04

Winter Term 2026/2027

Exercise 1

Which of the following literals describe the same value? Explain your answer.

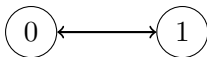
- a) `"2"^^xsd:integer` vs. `"2.0"^^xsd:decimal`
- b) `"2"^^xsd:decimal` vs. `"2"^^xsd:float`
- c) `"2018-11-06T15:40:00+01:00"^^xsd:dateTime` vs.
`"2018-11-06T14:40:00Z"^^xsd:dateTime`
- d) `"2018-11-06T15:40:00+01:00"^^xsd:dateTime` vs.
`"2018-11-06T14:40:00"^^xsd:dateTime`

A detailed description of each of the various XML Schema datatypes is given in the online specification: see <https://www.w3.org/TR/xmlschema11-2/>.

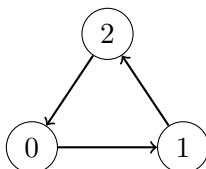
Exercise 2

A *graph homomorphism* from a directed graph $G = \langle V, E \rangle$ to a directed graph $G' = \langle V', E' \rangle$ is a map $h : V \rightarrow V'$ such that for every edge $\langle a, b \rangle \in E$, we also have $\langle h(a), h(b) \rangle \in E'$, i. e., such that adjacent vertices are mapped to adjacent vertices. A *strong homomorphism* furthermore requires that if $\langle h(a), h(b) \rangle \in E'$, then we already have $\langle a, b \rangle \in E$, i. e., only adjacent vertices are mapped to adjacent vertices.

For each of the following graphs, determine if there is a homomorphism from the graph to the graph $G' = \langle V', E' \rangle$ with $V' = \{0, 1\}$ and $E' = \{\langle 0, 1 \rangle, \langle 1, 0 \rangle\}$. Which graphs also admit a strong homomorphism to G' ?



- a) $G_1 = \langle V_1, E_1 \rangle$ with $V_1 = \{0, 1, 2\}$ and $E_1 = \{\langle 0, 1 \rangle, \langle 1, 2 \rangle, \langle 2, 0 \rangle\}$



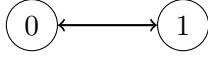
- b) $G_2 = \langle V_2, E_2 \rangle$ with $V_2 = \{0, 1\}$ and $E_2 = \emptyset$



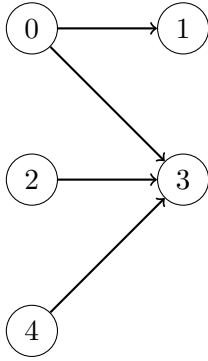
c) $G_3 = \langle V_3, E_3 \rangle$ with $V_3 = \{0\}$ and $E_3 = \{ \langle 0, 0 \rangle \}$



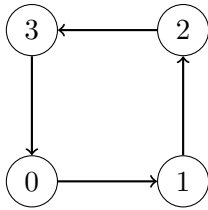
d) $G_4 = \langle V_4, E_4 \rangle$ with $V_4 = \{0, 1\}$ and $E_4 = \{ \langle 0, 1 \rangle, \langle 1, 0 \rangle \}$



e) $G_5 = \langle V_5, E_5 \rangle$ with $V_5 = \{0, 1, 2, 3, 4\}$ and $E_5 = \{ \langle 0, 1 \rangle, \langle 0, 3 \rangle, \langle 2, 3 \rangle, \langle 4, 3 \rangle \}$.



f) $G_6 = \langle V_6, E_6 \rangle$ with $V_6 = \{0, 1, 2, 3\}$ and $E_6 = \{ \langle 0, 1 \rangle, \langle 1, 2 \rangle, \langle 2, 3 \rangle, \langle 3, 0 \rangle \}$



Exercise 3

Recall that blank nodes act as placeholders for arbitrary resources in RDF: they assert that there is something without saying what it is. Such an assertion might logically follow from other, stronger assertions, so that some triples in a graph might be redundant. For example, the second triple in the following dataset can be omitted without loss of information:

```

eg:s    eg:p    eg:o .
_:1     eg:p    _:2 .
  
```

More generally, an *instance* of an RDF graph G is a graph $\sigma(G)$ obtained by applying a function σ that maps blank nodes to arbitrary RDF terms. A graph is *lean* if it does not have any instance $\sigma(G) \subset G$ that is strictly contained in G . In the example, $\sigma = \{ _ : 1 \mapsto \langle s \rangle, _ : 2 \mapsto \langle o \rangle \}$ shows that this graph is not lean.

Determine if the following graphs are lean:

- | | | | |
|-----|--|-----|---|
| (a) | eg:s eg:p eg:o .
_:1 eg:p _:1 . | (c) | eg:s eg:p eg:o .
_:1 eg:p [eg:p []] . |
| (b) | eg:s eg:p _:2 .
_:1 eg:p eg:o . | (d) | eg:s eg:p eg:s .
_:1 eg:p [eg:p []] . |

Exercise 4 (*)

Show that it is NP-complete to decide if an RDF graph is not lean.

Hint:

if261f

hard. Making it lean if it is not colourable requires some trick to prevent the encoded graph from embedding into $\mathbb{F}_2$. For hardness, find a reduction from 3-colourability. Making an RDF graph non-lean if a graph is colourable is not

Exercise 5

Write a program that reads a graph in N-Triples format and checks whether the graph is bipartite. Use it to decide whether `authorship.nt.gz`¹ and `coauthors.nt.gz`¹ are bipartite.

Hint: each of the uncompressed graphs is roughly 4 GiB in size. In Python, you can use `gzip.GzipFile`² to process the compressed file without decompressing it first. There is also `authorship-snippet.nt.gz`¹, a small part of the graph that you can use during development.

Please note: In order to get the correct data files, please install `git-lfs`³ on your system, and then activate it in your local repository (`git lfs install`).

¹<https://github.com/knownsys/Course-Knowledge-Graphs/tree/main/data/dblp>

²<https://docs.python.org/3/library/gzip.html>

³<https://git-lfs.github.com/>