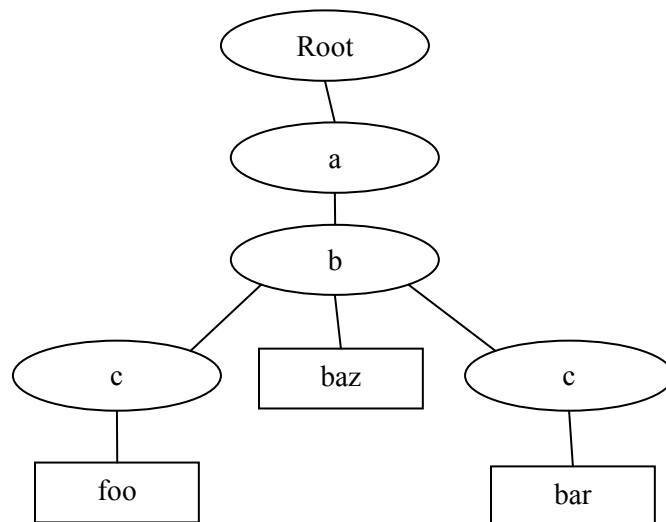


# Homework #1 Solutions

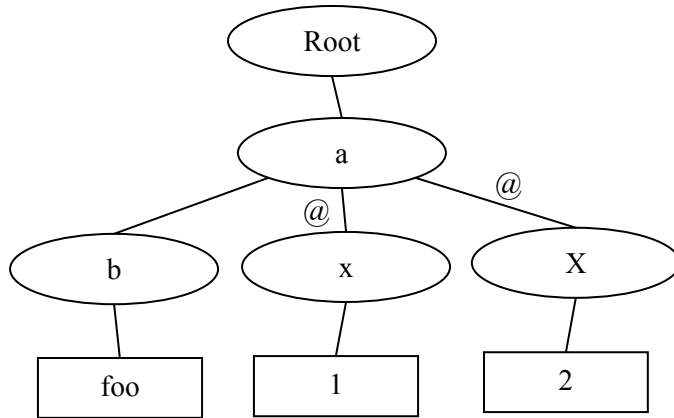
1. 3pts each

- a) `<a><b/><a>`  
*No, the last `<a>` should be `</a>` in order to close the element*
- b) `<a><b>foo</b></a><a>bar</a>`  
*No, this document has two roots*
- c) `<a><b>foo</b><b>bar</a>`  
*No, the second `<b>` should have a `</b>` tag*
- d) `<a><b><c>foo</b>bar</c></a>`  
*No, the b and c elements are not properly nested*
- e) `<a/><b>foo</b><b>bar<b>`  
*No, this document has no root.*
- f) `<a><b><c>foo</c>baz<c>bar</c></b></a>`

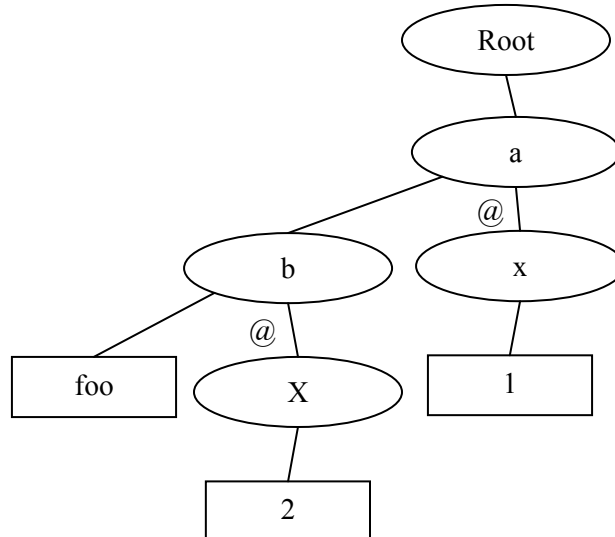


g) `<a x="1" X="2"><b>foo</b></a>`

*Note, since XML is case-sensitive, a's attributes are distinct*



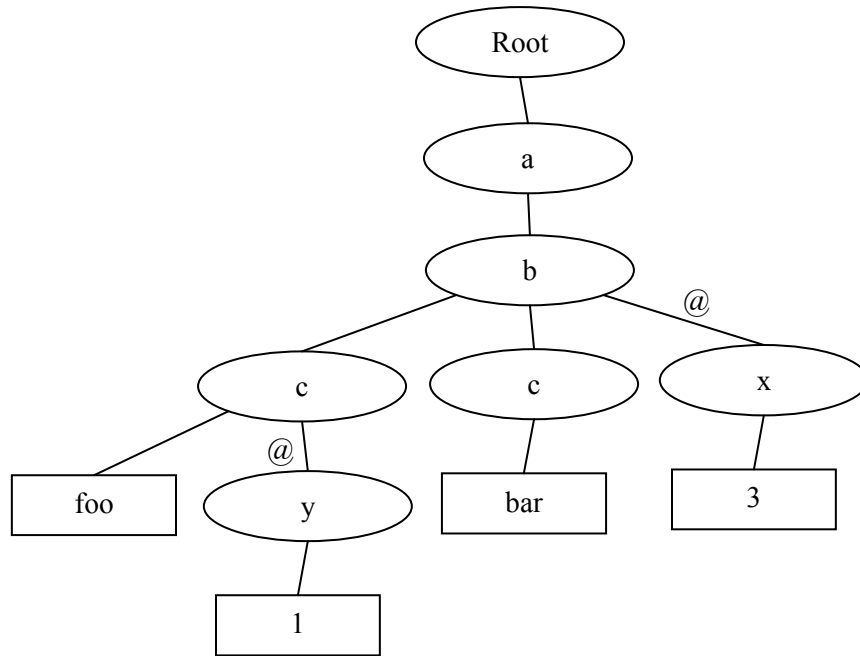
h) `<a x="1"><b x="2">foo</b></a>`



i) `<a x="1" y="2" x="3"><b>foo</b></a>`

*No, elements can't have duplicate attributes, and this has two x's*

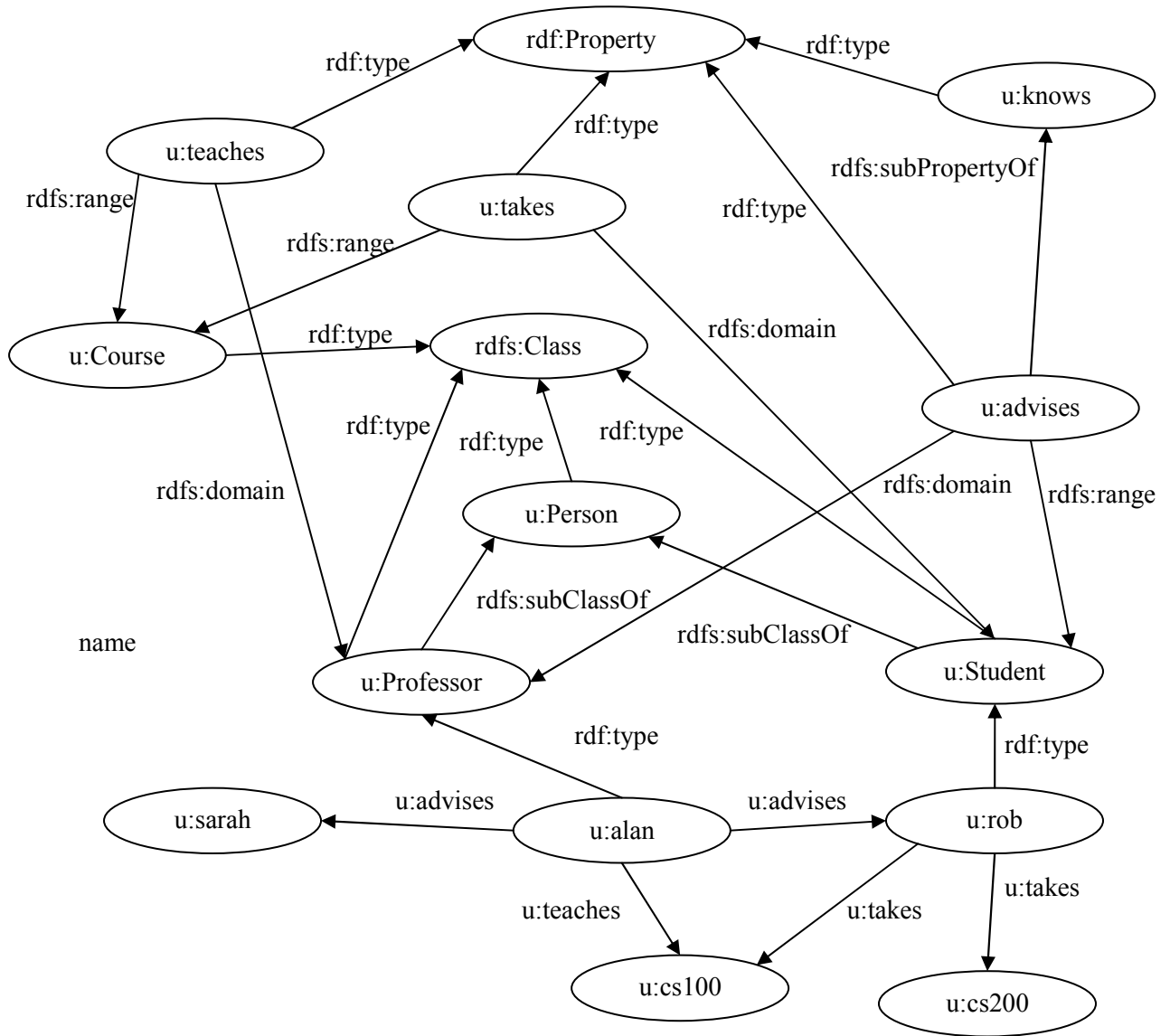
j) `<a><b x="3"><c y="1">foo</c><c>bar</c></b></a>`



2.

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
  <rdfs:Class rdf:ID="Person" />
  <rdfs:Class rdf:ID="Document" />
  <rdfs:Class rdf:ID="Book">
    <rdfs:subClassOf rdf:resource="#Document" />
  </rdfs:Class>
  <Person rdf:ID="king">
    <name>Stephen King</name>
    <wrote rdf:resource="#stand" />
  </Person>
  <Book rdf:ID="stand">
    <title>The Stand</title>
    <year>1978</year>
  </Book>
</rdf:RDF>
```

3.



4. New triples added: *(2 pts. each)*

u:sarah rdf:type u:Student (by rdfs3)

u:cs100 rdf:type u:Course (by rdfs3)

u:cs200 rdf:type u:Course (by rdfs3)

u:alan u:knows u:rob (by rdfs7)

u:alan u:knows u:sarah (by rdfs7)

u:alan rdf:type u:Person (by rdfs9)

u:rob rdf:type u:Person (by rdfs9)

u:sarah rdf:type u:Person (by rdfs9)

Others given by a rule but already in document

u:rob rdf:type u:Student (by rdfs2)

u:alan rdf:type u:Professor (by rdfs2)

u:rob rdf:type u:Student (by rdfs3)

5. Intersection semantics for rdfs:range make it possible to infer the rdf:type of resources that participate as the object in triples using the property. If there are multiple rdfs:range statements, then each one implies a new type for the resource (since the type of the resource must be of the intersection of all ranges). If new rdfs:range information is discovered, there is no need to retract information. On the other hand, if union semantics were used, one cannot infer anything. Even if only a single range triple was known for the property, it may be possible that there are unknown range triples, and any resource that was an object of a triple using the property could be of the type of one of the unknown ranges, instead of the known one.